FIG. 1

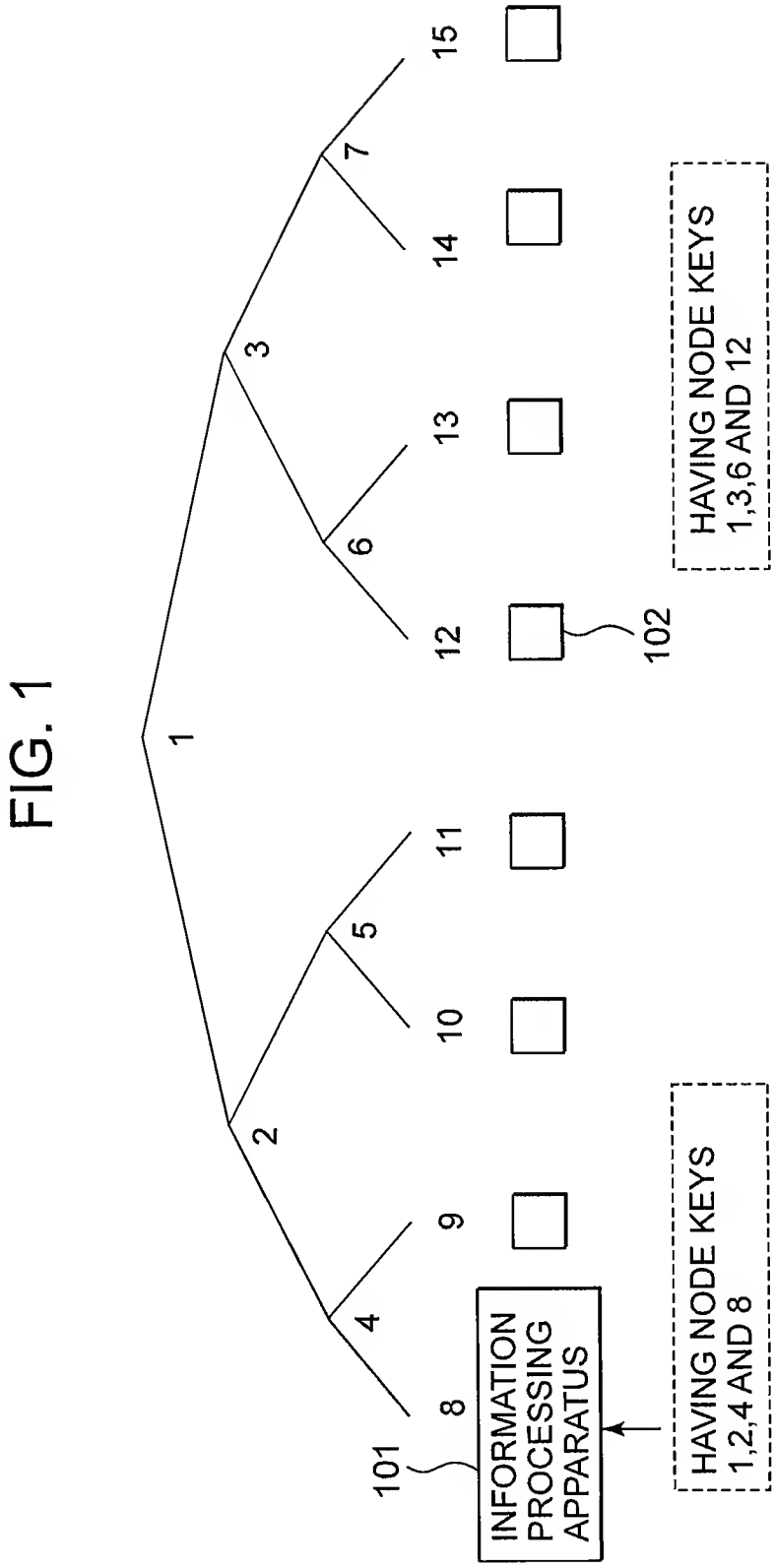

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
101
102
INFORMATION PROCESSING APPARATUS
HAVING NODE KEYS 1,2,4 AND 8
HAVING NODE KEYS 1,3,6 AND 12

FIG. 2

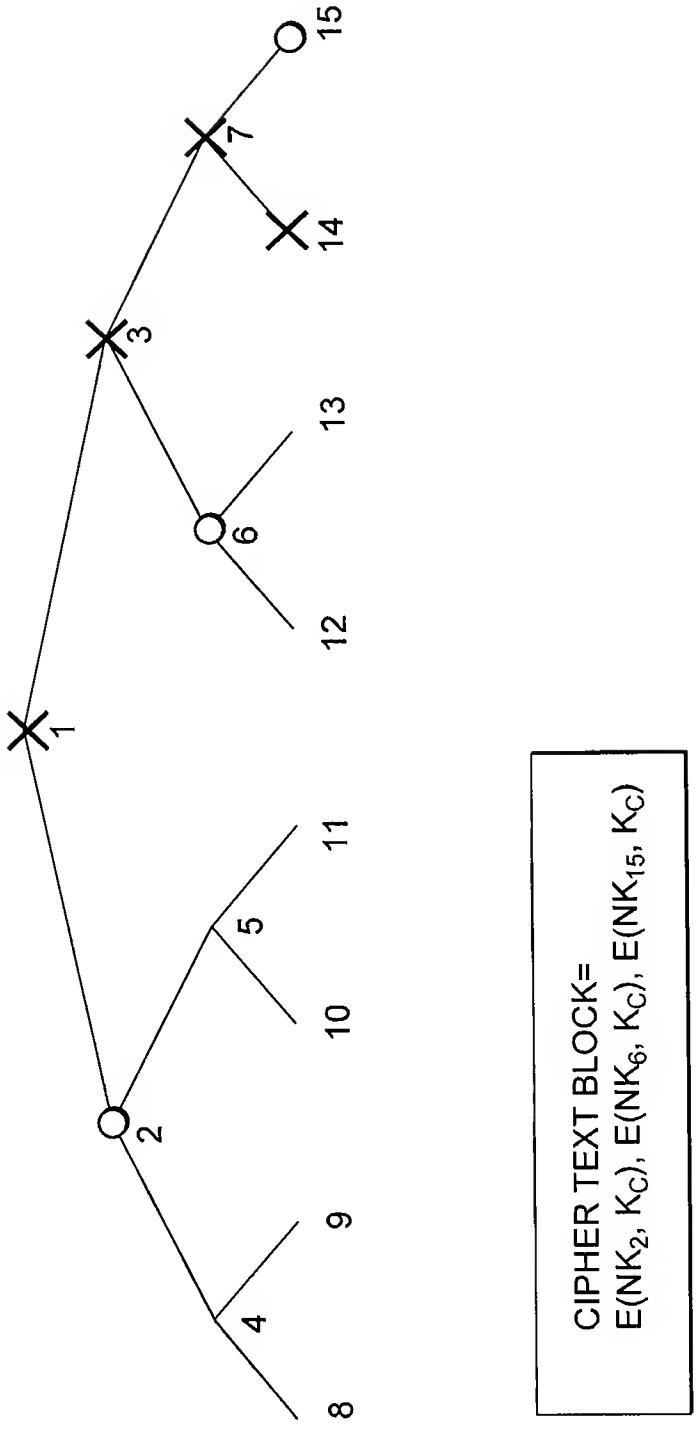
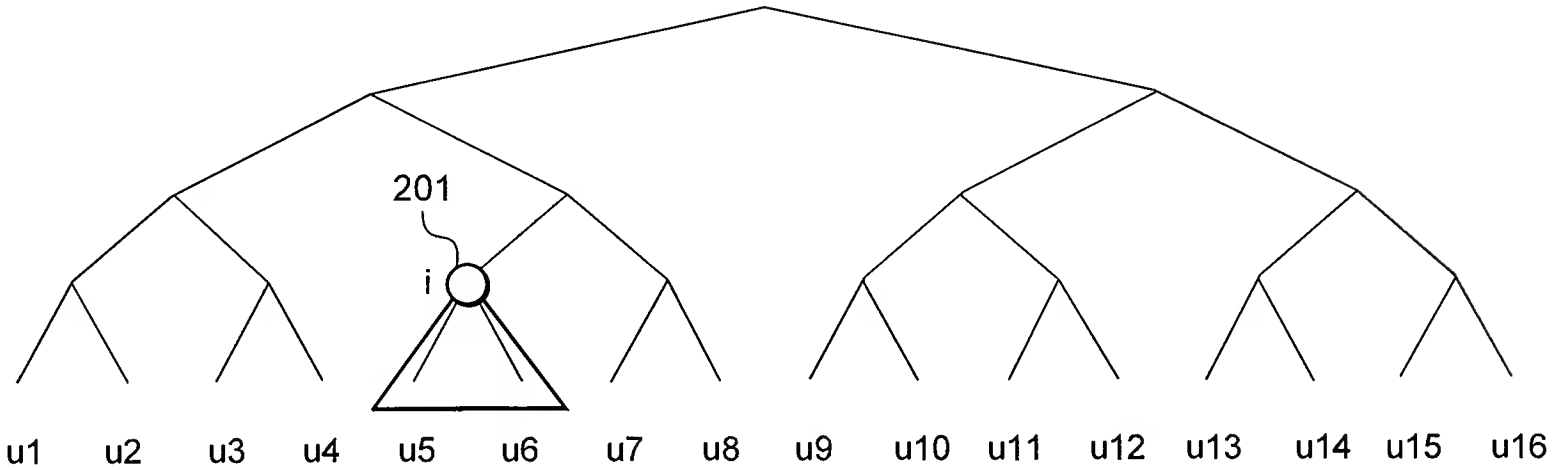

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
CIPHER TEXT BLOCK=
E(NK2, KC), E(NK6, KC), E(NK15, KC)

FIG. 3

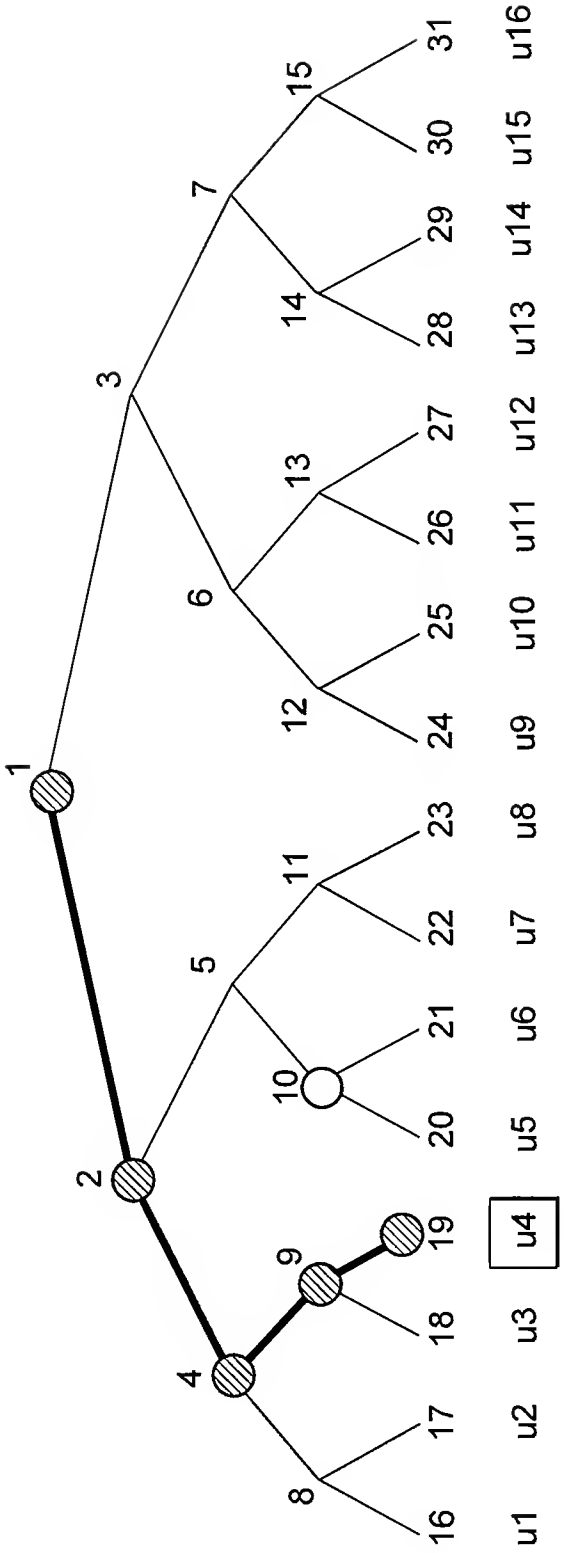


A "NODE" IS USED TO REPRESENT "SET CONSISTING OF LEAVES OF SUBTREE ROOTED AT THE NODE"
Ex) Node i == Subset i (S_i) == {u5, u6}

SUCH SET IS DEFINED AS TO ALL NODES OF TREE

NUMBER OF SUBSETS TO WHICH CERTAIN RECEIVER BELONGS = NUMBER OF KEYS WHICH EACH RECEIVER HOLDS= log N + 1

FIG. 4



NODE KEYS OWNED BY u4: NODE KEYS FOR NODES 1, 2, 4, 9 AND 19

FIG. 5

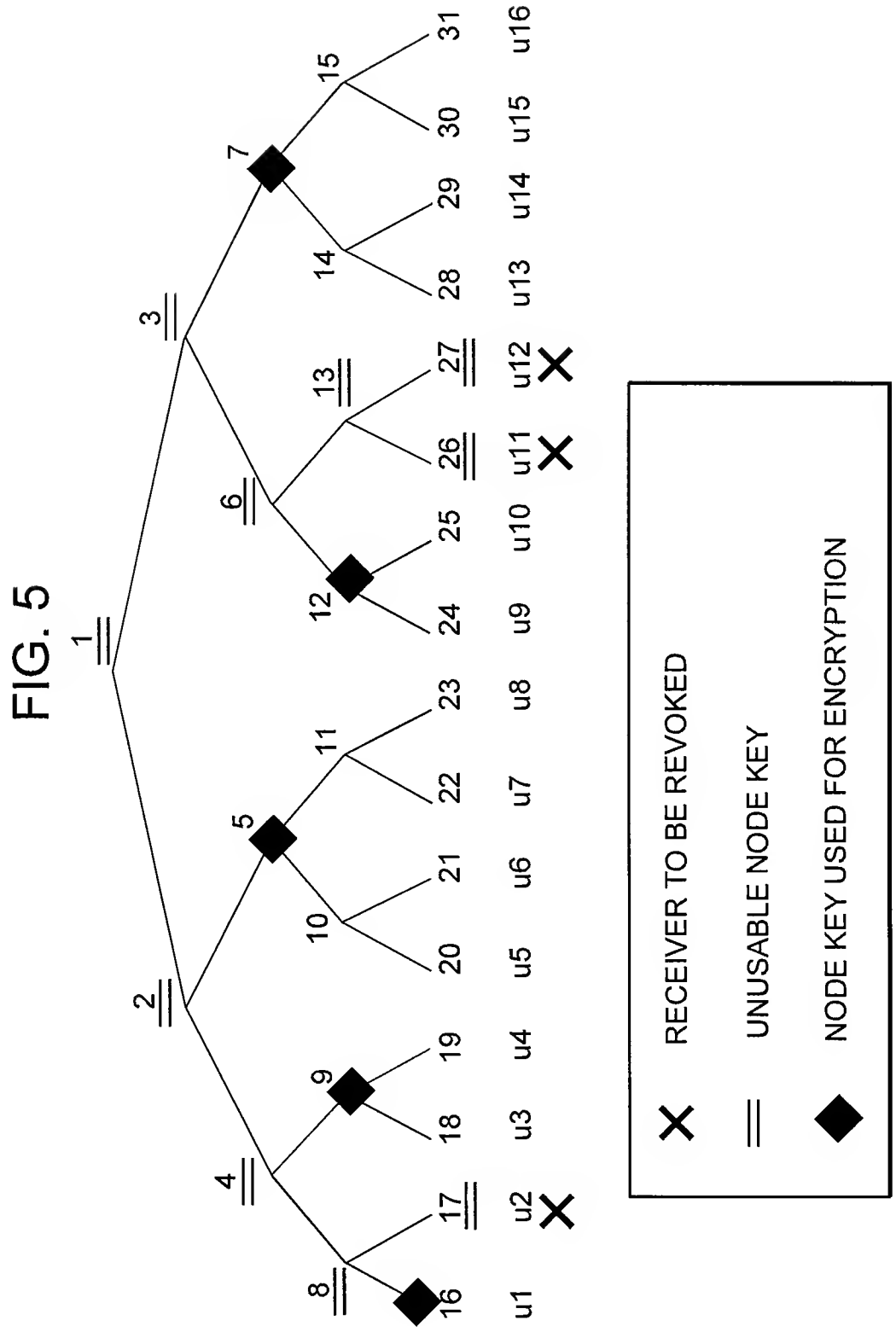
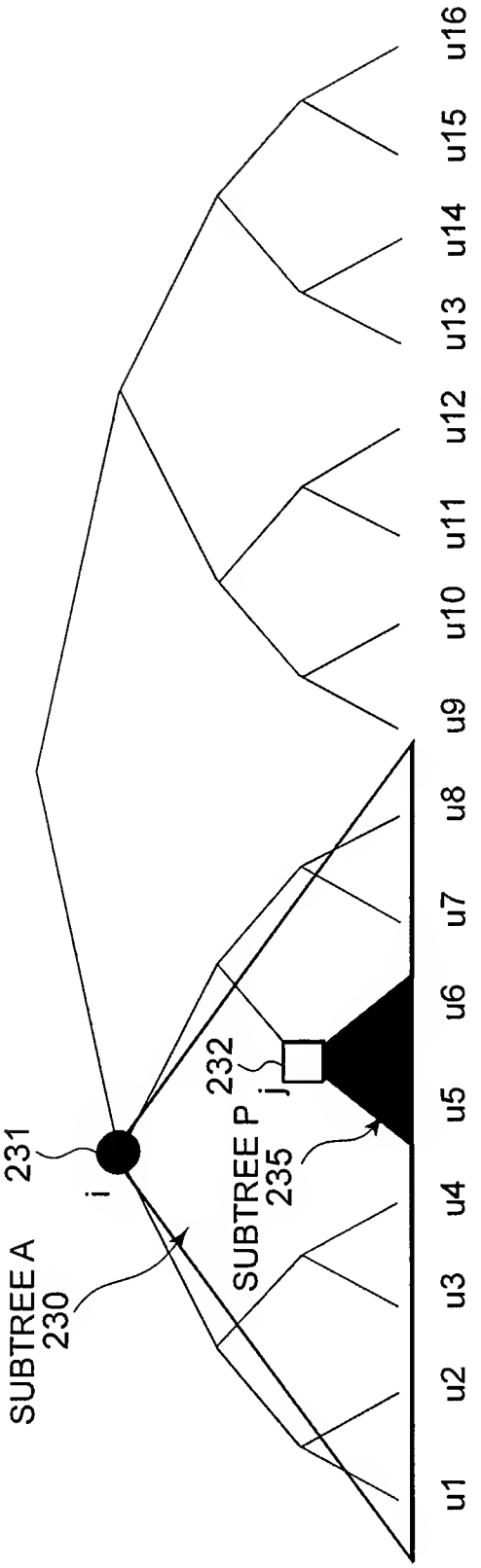
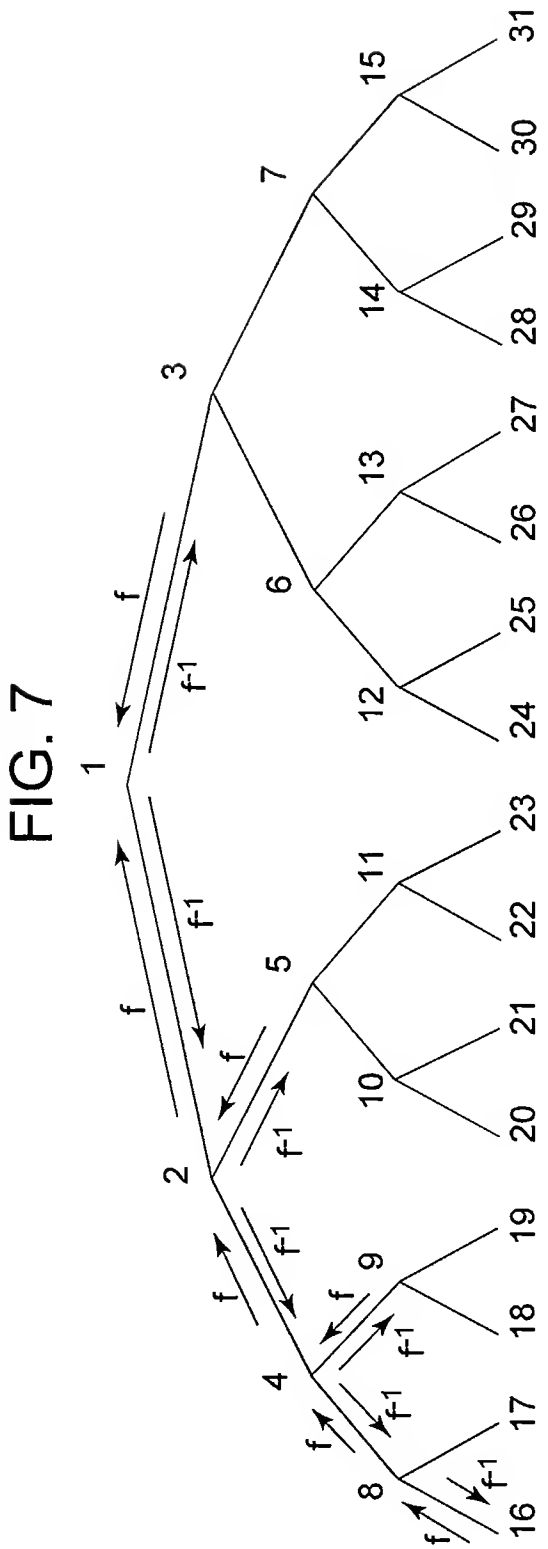

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
u1
u2
u3
u4
u5
u6
u7
u8
u9
u10
u11
u12
u13
u14
u15
u16
RECEIVER TO BE REVOKED
UNUSABLE NODE KEY
NODE KEY USED FOR ENCRYPTION

FIG. 6



WHEN NODE i IS ANCESTOR OF NODE j
RECEIVERS (u5, u6) HAVING NODE KEY FOR NODE j ALWAYS
HAS NODE KEY FOR NODE i

FIG. 7



f: COMPUTATION USING FORWARD PERMUTATION F OF RSA

f^{-1}: COMPUTATION USING INVERSE PERMUTATION F^{-1} OF RSA

FIG. 8

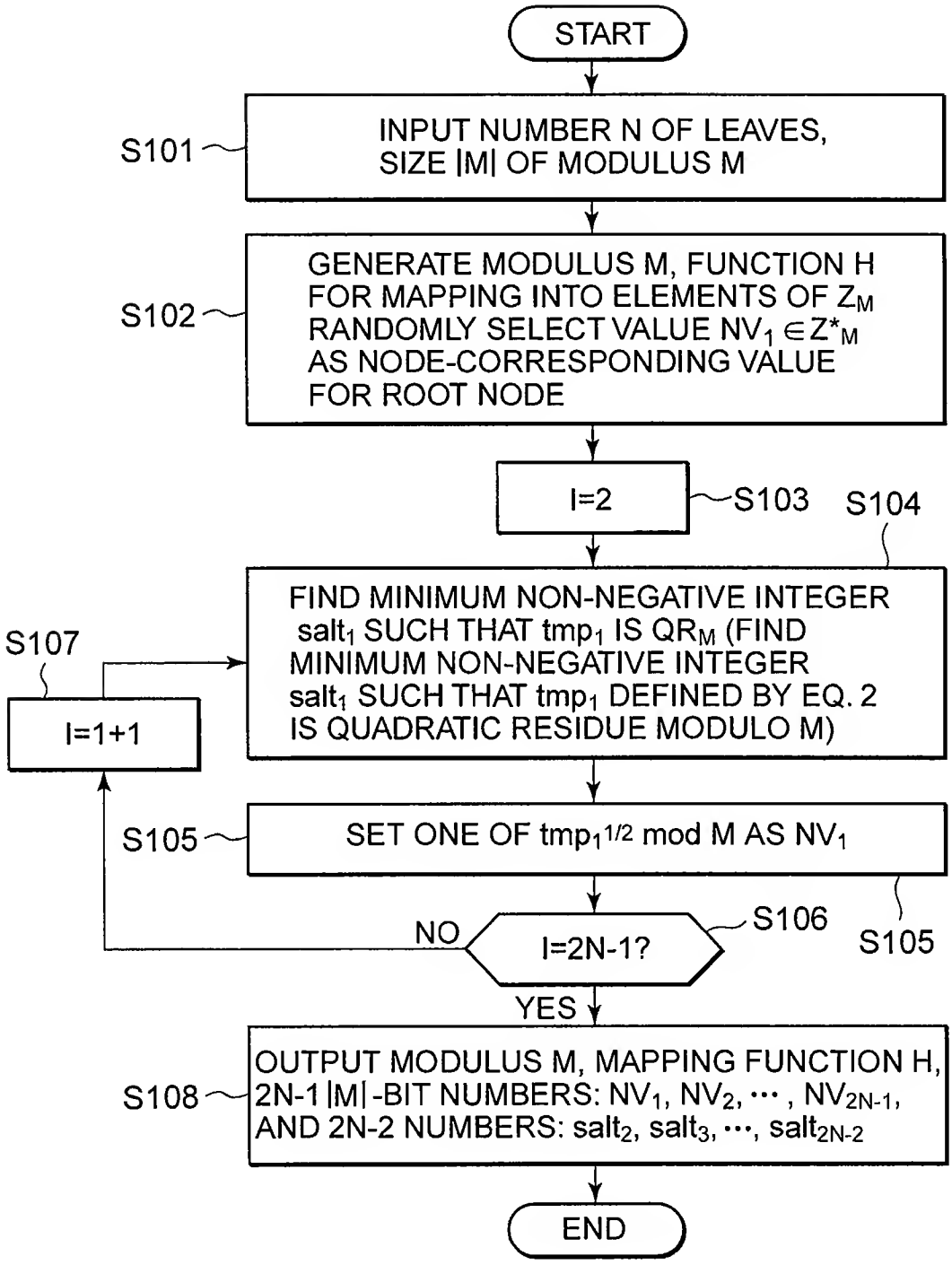
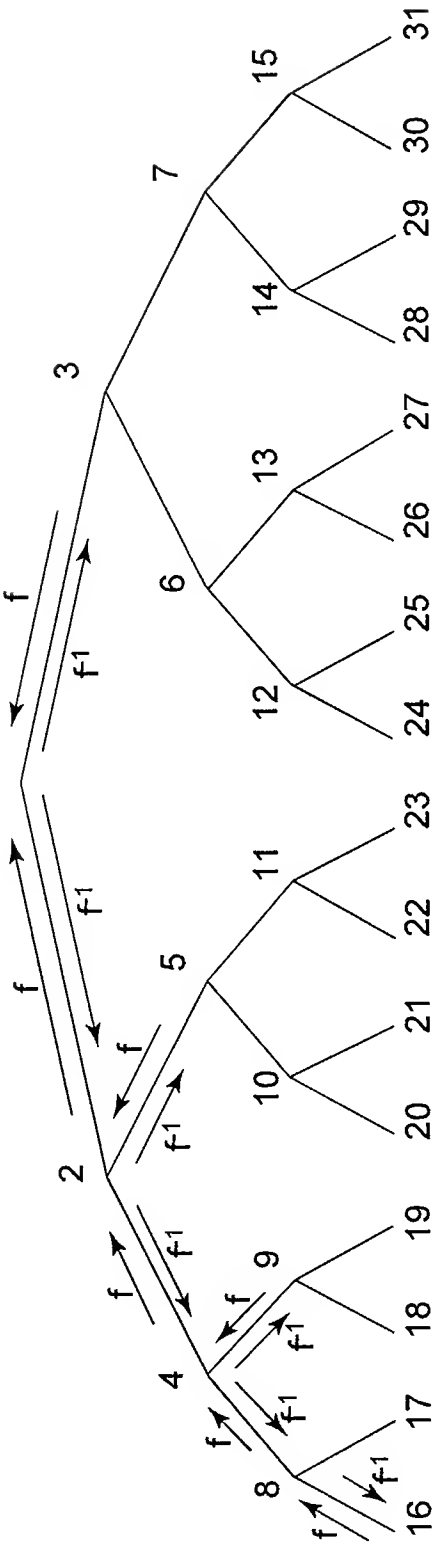
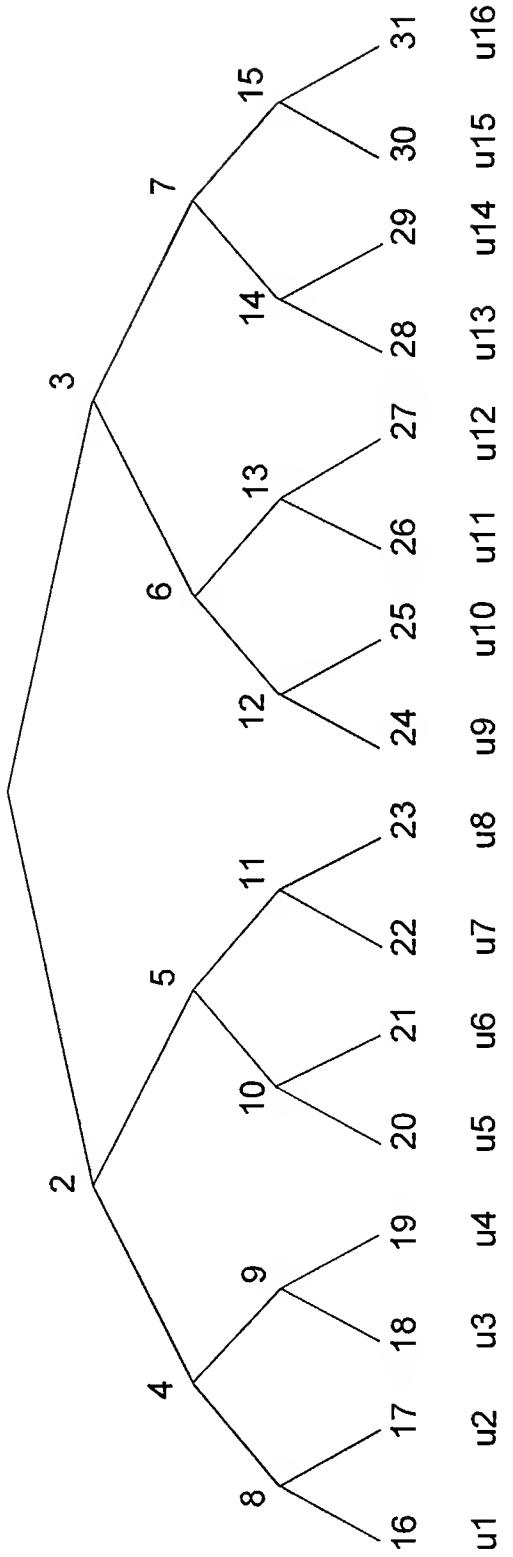

START
S101
INPUT NUMBER N OF LEAVES, SIZE |M| OF MODULUS M
S102
GENERATE MODULUS M, FUNCTION H FOR MAPPING INTO ELEMENTS OF Z_M RANDOMLY SELECT VALUE $NV_1 \in Z^*_M$ AS NODE-CORRESPONDING VALUE FOR ROOT NODE
I=2
S103
S104
FIND MINIMUM NON-NEGATIVE INTEGER $salt_1$ SUCH THAT tmp_1 IS QR_M (FIND MINIMUM NON-NEGATIVE INTEGER $salt_1$ SUCH THAT tmp_1 DEFINED BY EQ. 2 IS QUADRATIC RESIDUE MODULO M)
S107
I=1+1
S105
SET ONE OF $tmp_1^{1/2}$ mod M AS NV_1
NO
I=2N-1?
S106
S105
YES
S108
OUTPUT MODULUS M, MAPPING FUNCTION H, 2N-1 |M|-BIT NUMBERS: NV_1, NV_2, ⋯ , NV_{2N-1}, AND 2N-2 NUMBERS: $salt_2$, $salt_3$, ⋯, $salt_{2N-2}$
END

FIG. 9



f: COMPUTATION USING FORWARD COMPUTATION (SQUARING MODULO M) F
f^{-1}: COMPUTATION USING INVERSE COMPUTATION (FINDING SQUARE ROOTS MODULO M) F^{-1}

FIG. 10



RECEIVER u4 IS GIVEN NV19
AND salt19, salt9, salt4, salt2

FIG. 11

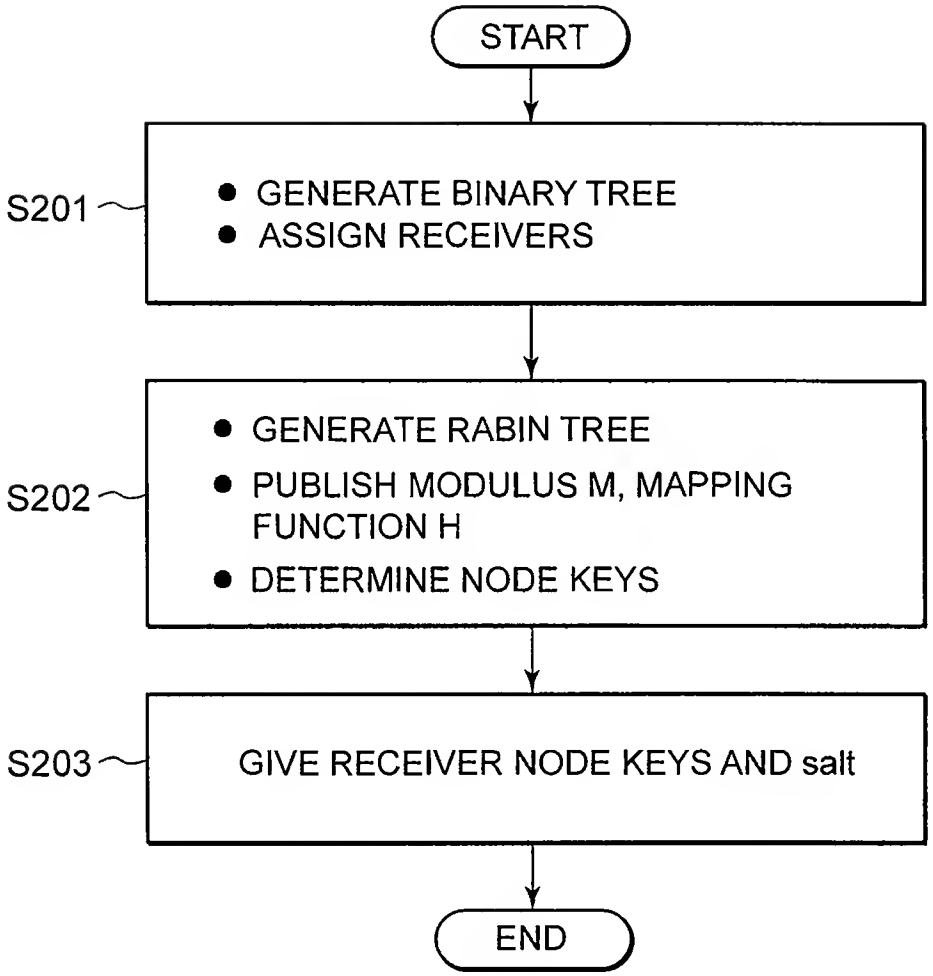

START
S201
GENERATE BINARY TREE
ASSIGN RECEIVERS
S202
GENERATE RABIN TREE
PUBLISH MODULUS M, MAPPING FUNCTION H
DETERMINE NODE KEYS
S203
GIVE RECEIVER NODE KEYS AND salt
END

FIG. 12

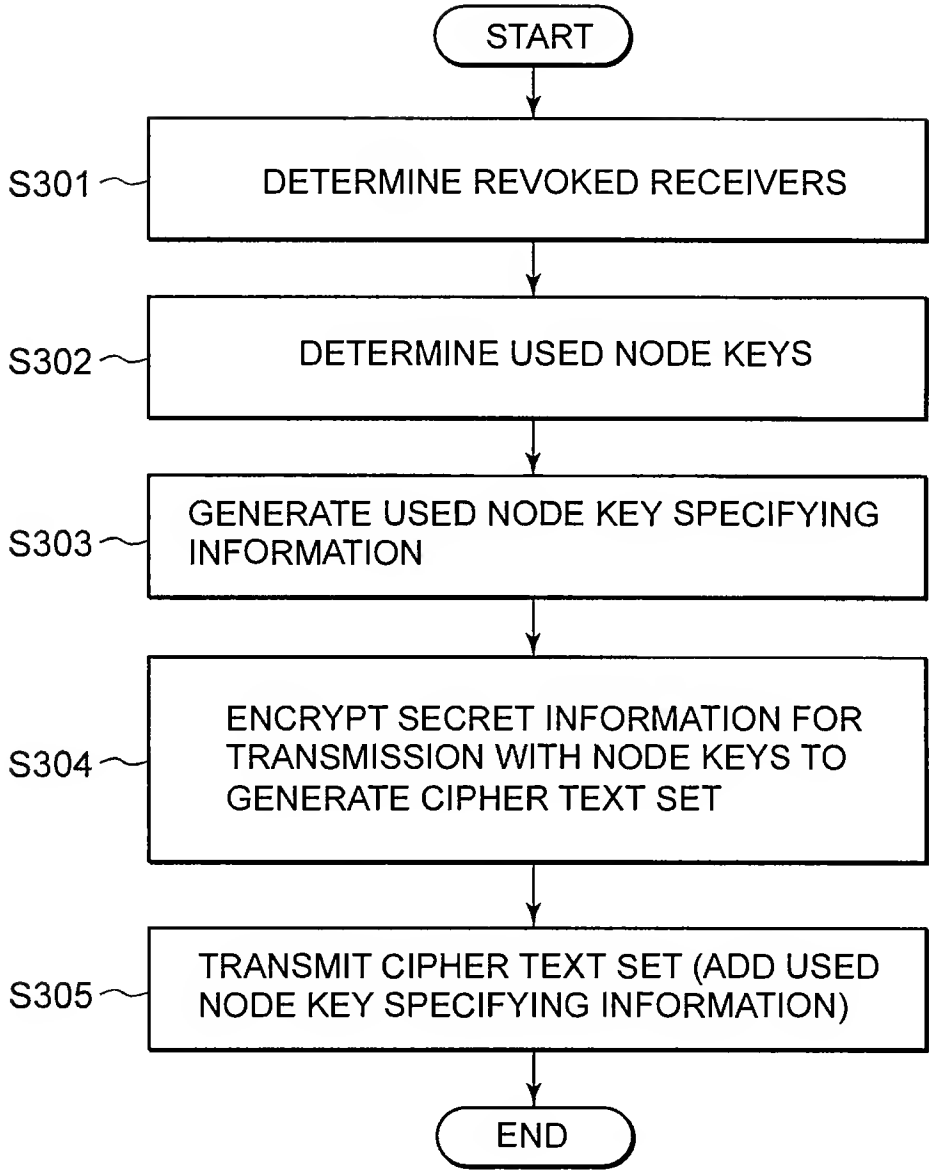
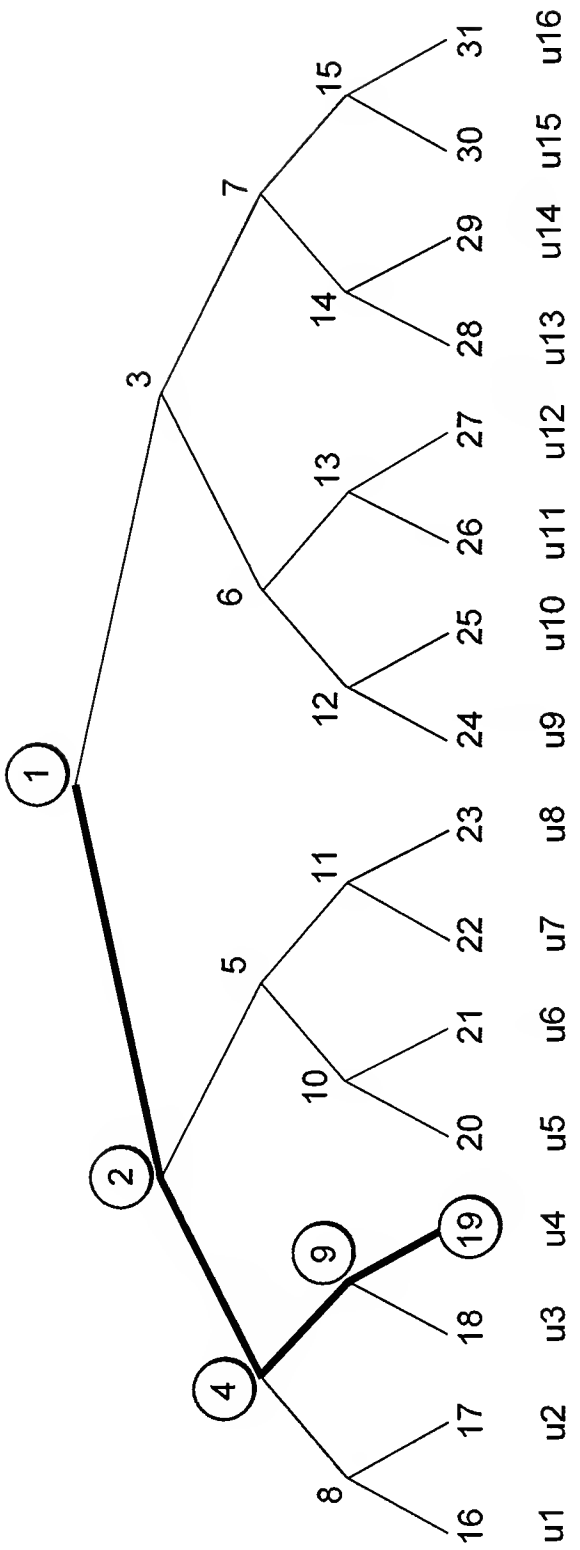

START
S301
DETERMINE REVOKED RECEIVERS
S302
DETERMINE USED NODE KEYS
S303
GENERATE USED NODE KEY SPECIFYING INFORMATION
S304
ENCRYPT SECRET INFORMATION FOR TRANSMISSION WITH NODE KEYS TO GENERATE CIPHER TEXT SET
S305
TRANSMIT CIPHER TEXT SET (ADD USED NODE KEY SPECIFYING INFORMATION)
END

FIG. 13



RECEIVER u4 IS GIVEN NV19
AND salt19, salt9, salt4, salt2

FIG. 14

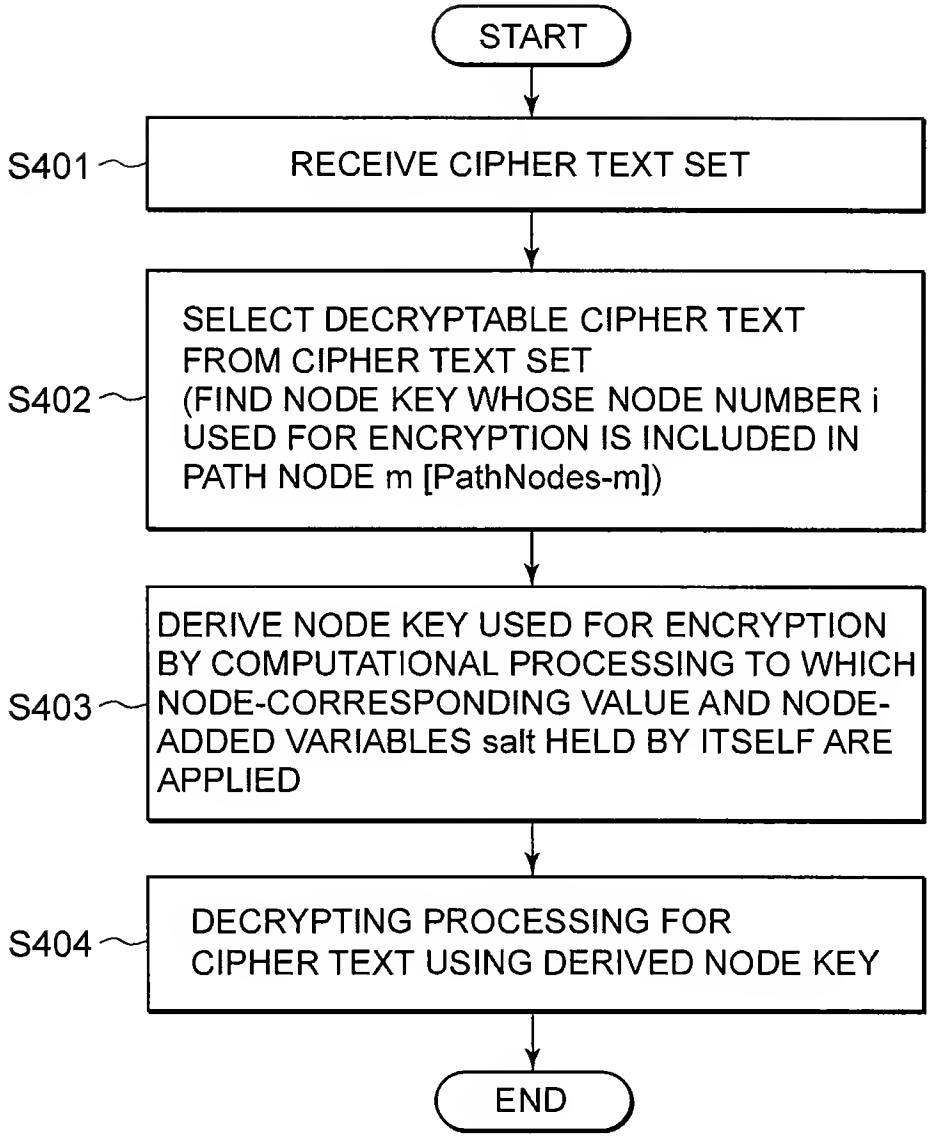

START
S401
RECEIVE CIPHER TEXT SET
S402
SELECT DECRYPTABLE CIPHER TEXT FROM CIPHER TEXT SET (FIND NODE KEY WHOSE NODE NUMBER i USED FOR ENCRYPTION IS INCLUDED IN PATH NODE m [PathNodes-m])
S403
DERIVE NODE KEY USED FOR ENCRYPTION BY COMPUTATIONAL PROCESSING TO WHICH NODE-CORRESPONDING VALUE AND NODE-ADDED VARIABLES salt HELD BY ITSELF ARE APPLIED
S404
DECRYPTING PROCESSING FOR CIPHER TEXT USING DERIVED NODE KEY
END

FIG. 15

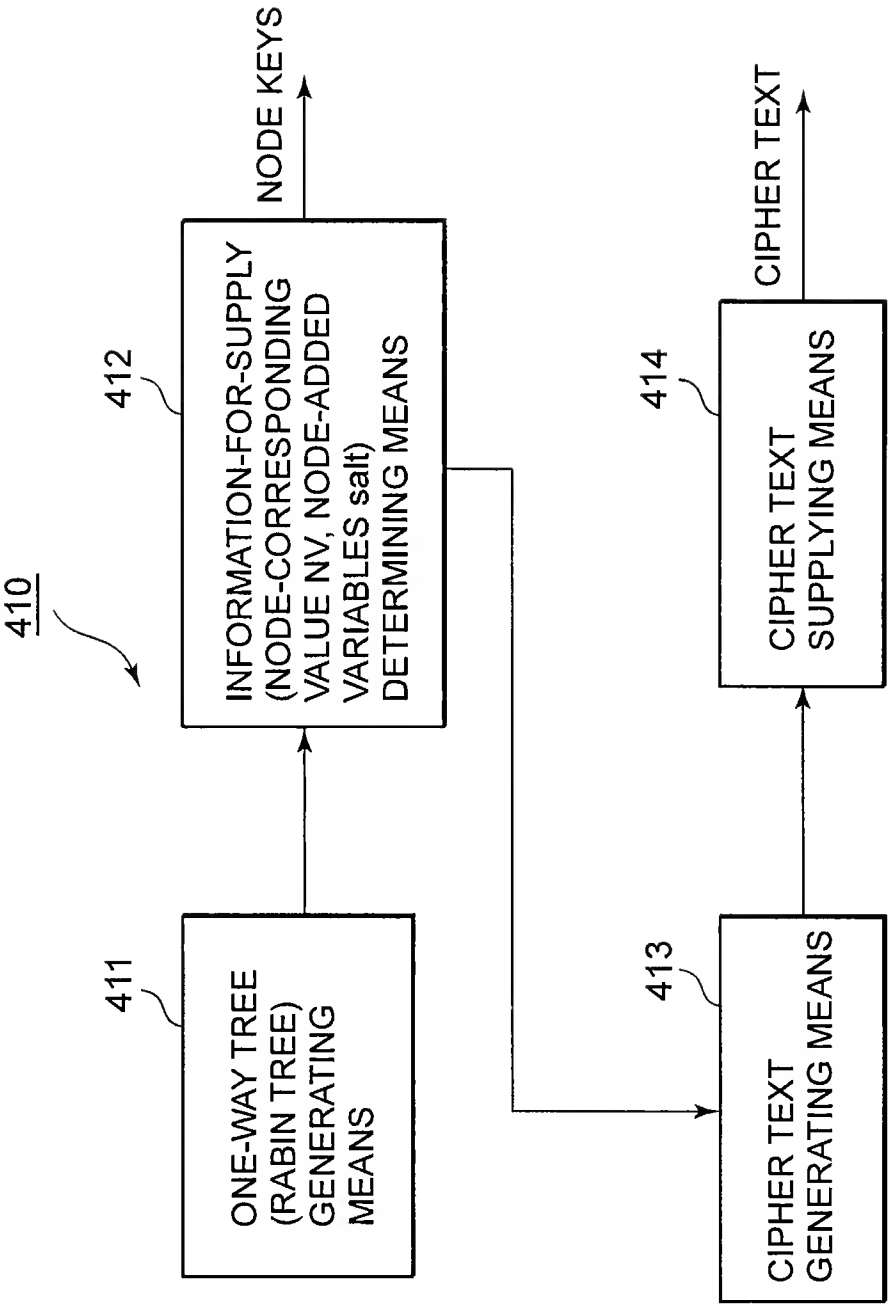

410
411
ONE-WAY TREE (RABIN TREE) GENERATING MEANS
412
INFORMATION-FOR-SUPPLY (NODE-CORRESPONDING VALUE NV, NODE-ADDED VARIABLES salt) DETERMINING MEANS
NODE KEYS
413
CIPHER TEXT GENERATING MEANS
414
CIPHER TEXT SUPPLYING MEANS
CIPHER TEXT

FIG. 16

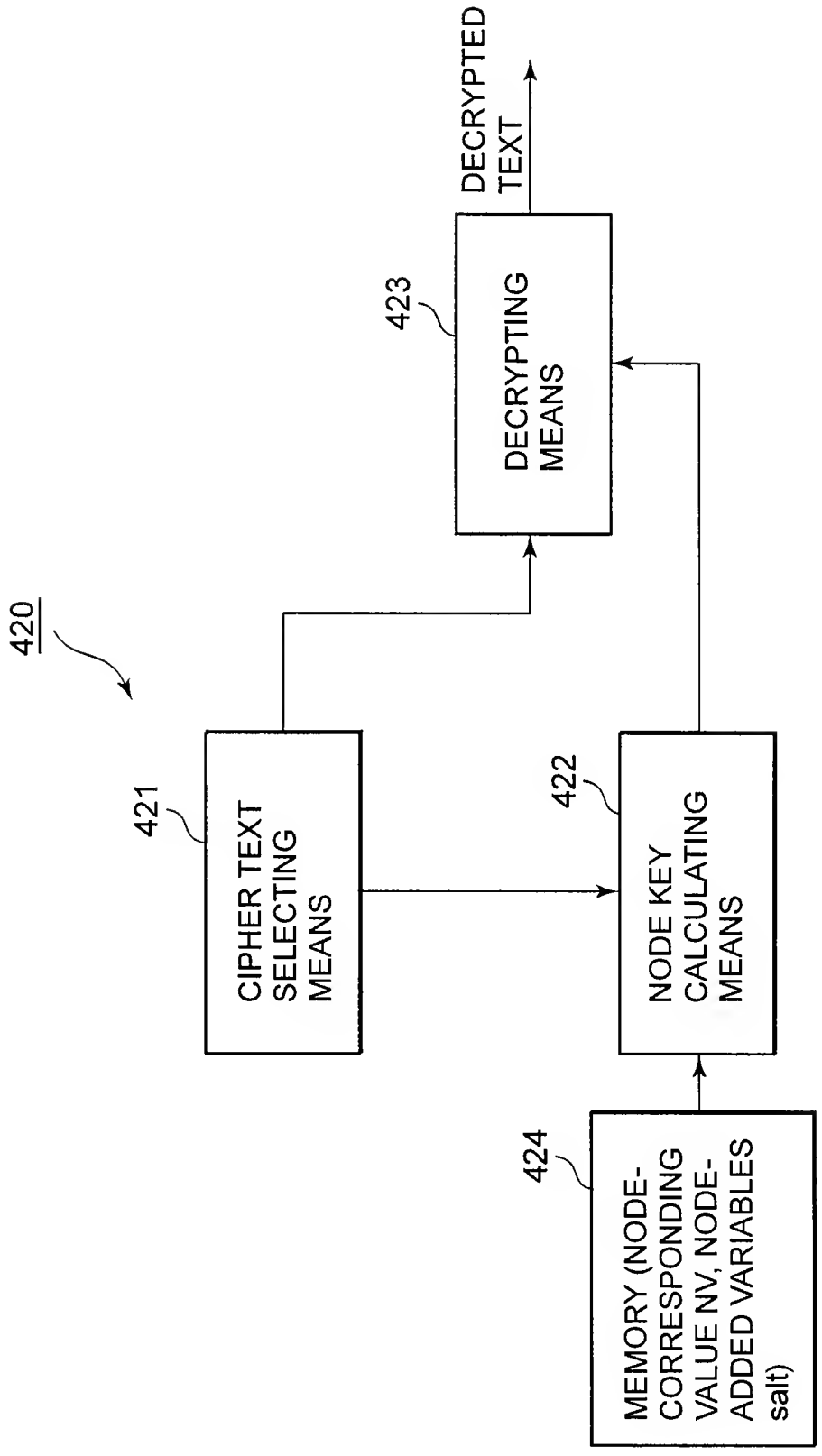

420
421
CIPHER TEXT SELECTING MEANS
422
NODE KEY CALCULATING MEANS
423
DECRYPTING MEANS
DECRYPTED TEXT
424
MEMORY (NODE-CORRESPONDING VALUE NV, NODE-ADDED VARIABLES salt)

FIG. 17

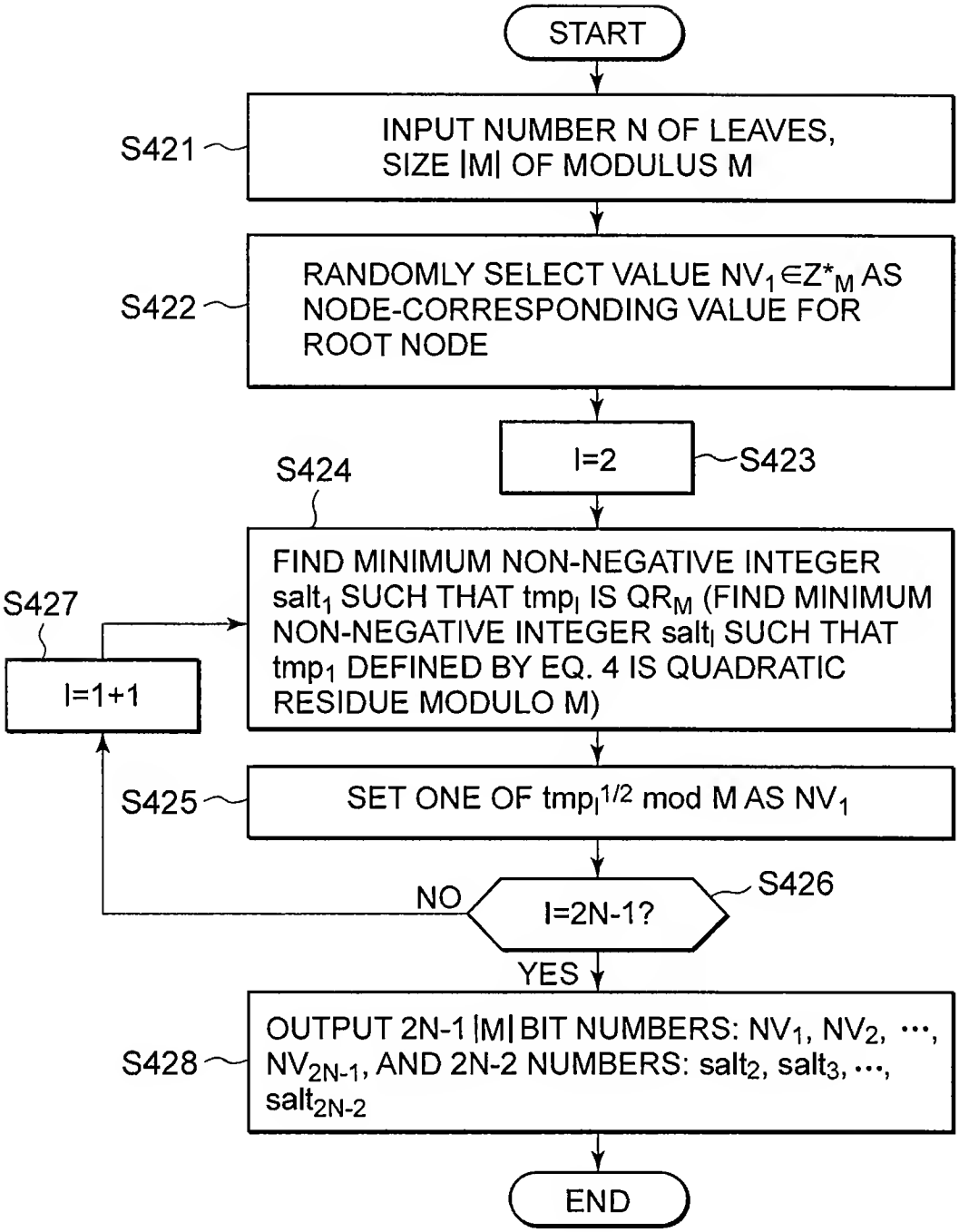
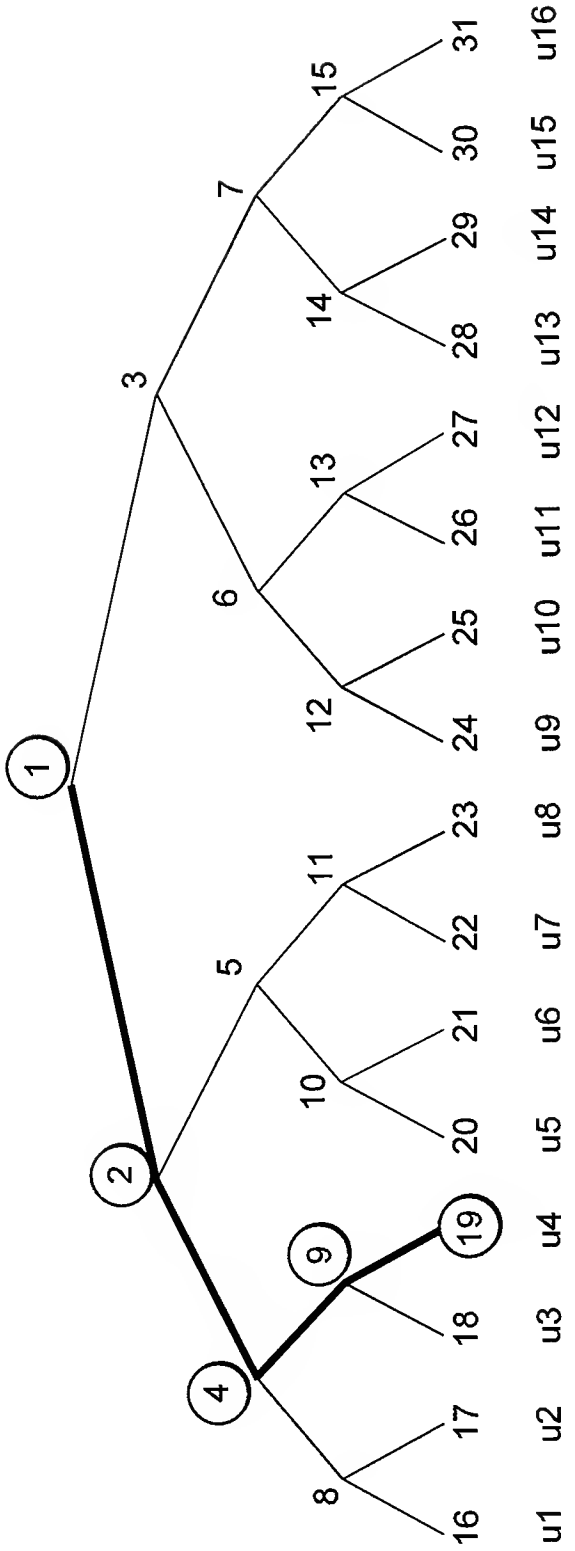

START
S421
INPUT NUMBER N OF LEAVES, SIZE |M| OF MODULUS M
S422
RANDOMLY SELECT VALUE $NV_1 \in Z^*_M$ AS NODE-CORRESPONDING VALUE FOR ROOT NODE
I=2
S423
S424
FIND MINIMUM NON-NEGATIVE INTEGER $salt_1$ SUCH THAT tmp_I IS QR_M (FIND MINIMUM NON-NEGATIVE INTEGER $salt_I$ SUCH THAT tmp_1 DEFINED BY EQ. 4 IS QUADRATIC RESIDUE MODULO M)
S427
I=1+1
S425
SET ONE OF $tmp_I^{1/2}$ mod M AS NV_1
NO
I=2N-1?
S426
YES
S428
OUTPUT 2N-1 |M| BIT NUMBERS: NV_1, NV_2, ···, NV_{2N-1}, AND 2N-2 NUMBERS: $salt_2$, $salt_3$, ···, $salt_{2N-2}$
END

FIG. 18



RECEIVER u4 IS GIVEN NV19
AND salt19, salt9, salt4, salt2

FIG. 19

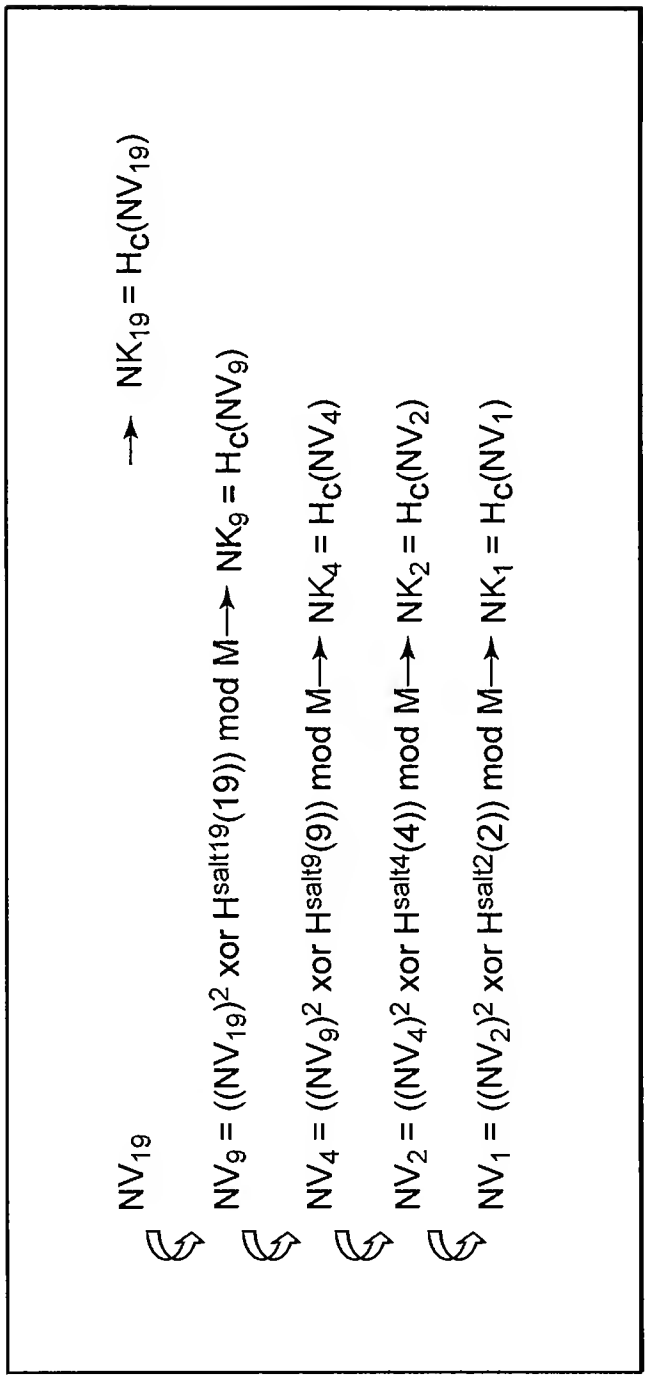

NV_{19}
$NV_9 = ((NV_{19})^2 \text{ xor } H^{salt19}(19)) \bmod M \rightarrow NK_9 = H_C(NV_9)$
$NV_4 = ((NV_9)^2 \text{ xor } H^{salt9}(9)) \bmod M \rightarrow NK_4 = H_C(NV_4)$
$NV_2 = ((NV_4)^2 \text{ xor } H^{salt4}(4)) \bmod M \rightarrow NK_2 = H_C(NV_2)$
$NV_1 = ((NV_2)^2 \text{ xor } H^{salt2}(2)) \bmod M \rightarrow NK_1 = H_C(NV_1)$
$\rightarrow NK_{19} = H_C(NV_{19})$

FIG. 20

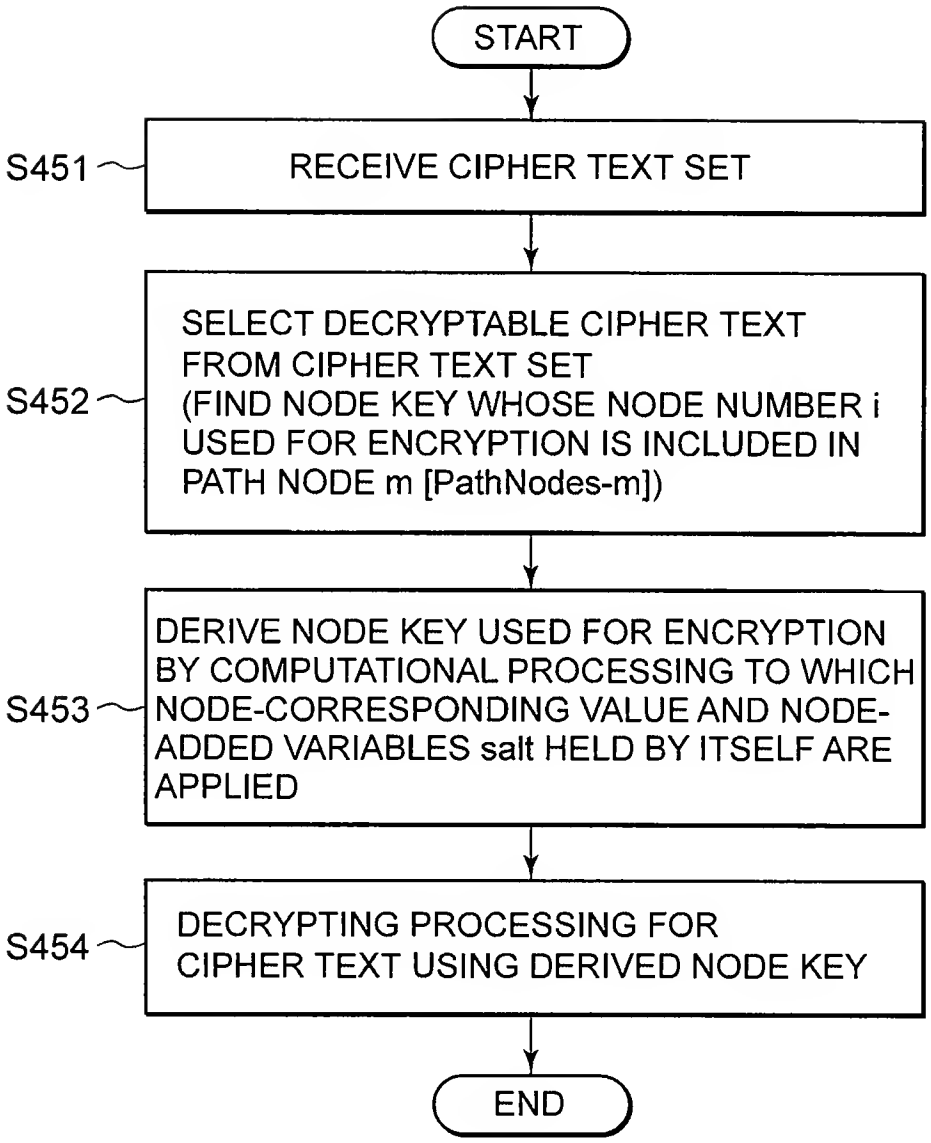
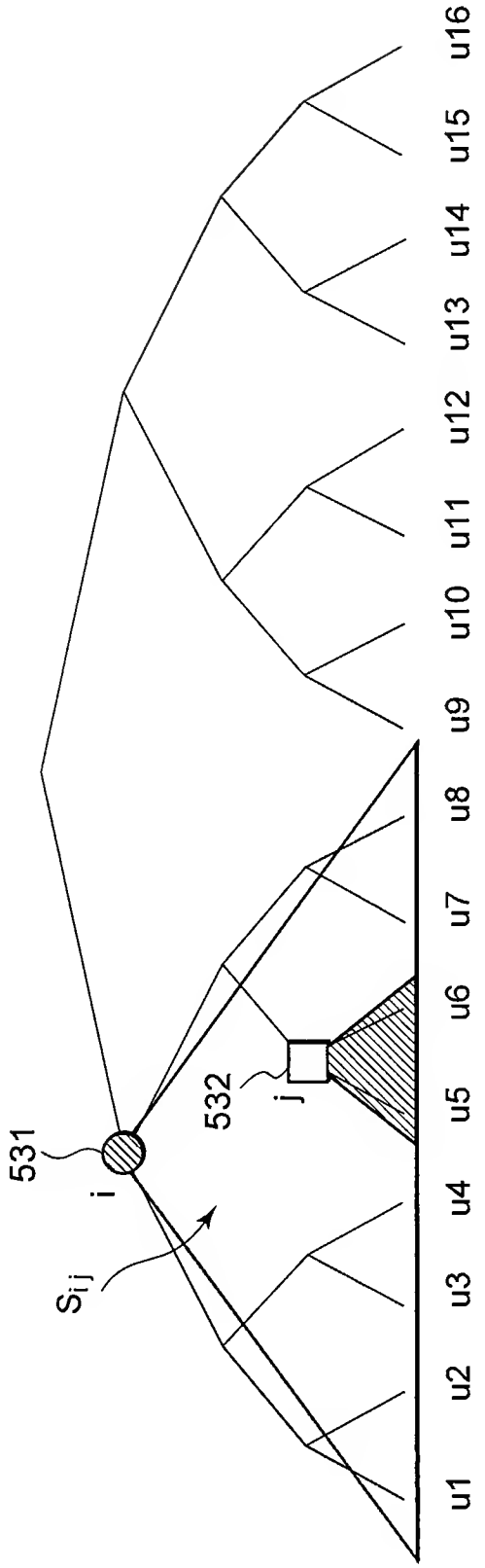

START
S451
RECEIVE CIPHER TEXT SET
S452
SELECT DECRYPTABLE CIPHER TEXT FROM CIPHER TEXT SET (FIND NODE KEY WHOSE NODE NUMBER i USED FOR ENCRYPTION IS INCLUDED IN PATH NODE m [PathNodes-m])
S453
DERIVE NODE KEY USED FOR ENCRYPTION BY COMPUTATIONAL PROCESSING TO WHICH NODE-CORRESPONDING VALUE AND NODE-ADDED VARIABLES salt HELD BY ITSELF ARE APPLIED
S454
DECRYPTING PROCESSING FOR CIPHER TEXT USING DERIVED NODE KEY
END

FIG. 21



"TWO NODES" ARE USED TO REPRESENT "SET CONSISTING OF LEAVES OF SUBTREE ROOTED AT FIRST NODE - SET CONSISTING OF LEAVES OF SUBTREE ROOTED AT SECOND NODE"
Ex) Node i,j == Subset i,j ($S_{i,j}$) == {u1,..., u8} - {u5, u6} = {u1, u2, u3, u4, u7, u8}

SUCH SET IS DEFINED AS TO ALL NODE PAIRS (i,j) WHERE i IS ANCESTOR OF j

FIG. 22

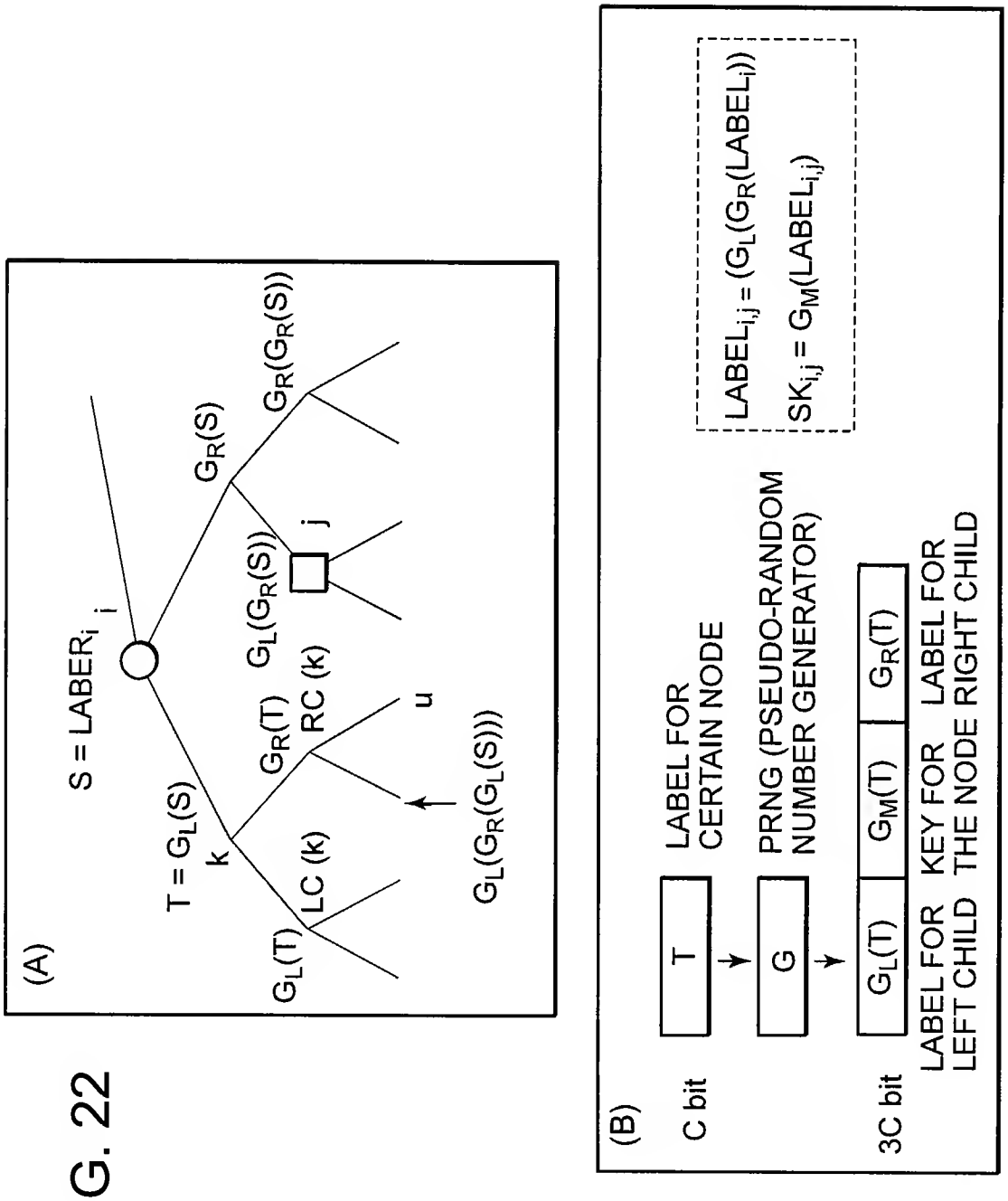

(A)
S = LABER$_i$ i
T = G_L(S)
k
G_L(T)
LC (k)
G_R(T)
RC (k)
G_L(G_R(S))
j
G_R(S)
G_R(G_R(S))
u
G_L(G_R(G_L(S)))
(B)
C bit
T
LABEL FOR CERTAIN NODE
G
PRNG (PSEUDO-RANDOM NUMBER GENERATOR)
3C bit
G_L(T)
G_M(T)
G_R(T)
LABEL FOR LEFT CHILD
KEY FOR THE NODE
LABEL FOR RIGHT CHILD
$LABEL_{i,j}$ = (G_L(G_R($LABEL_i$))
$SK_{i,j}$ = G_M($LABEL_{i,j}$)

FIG. 23

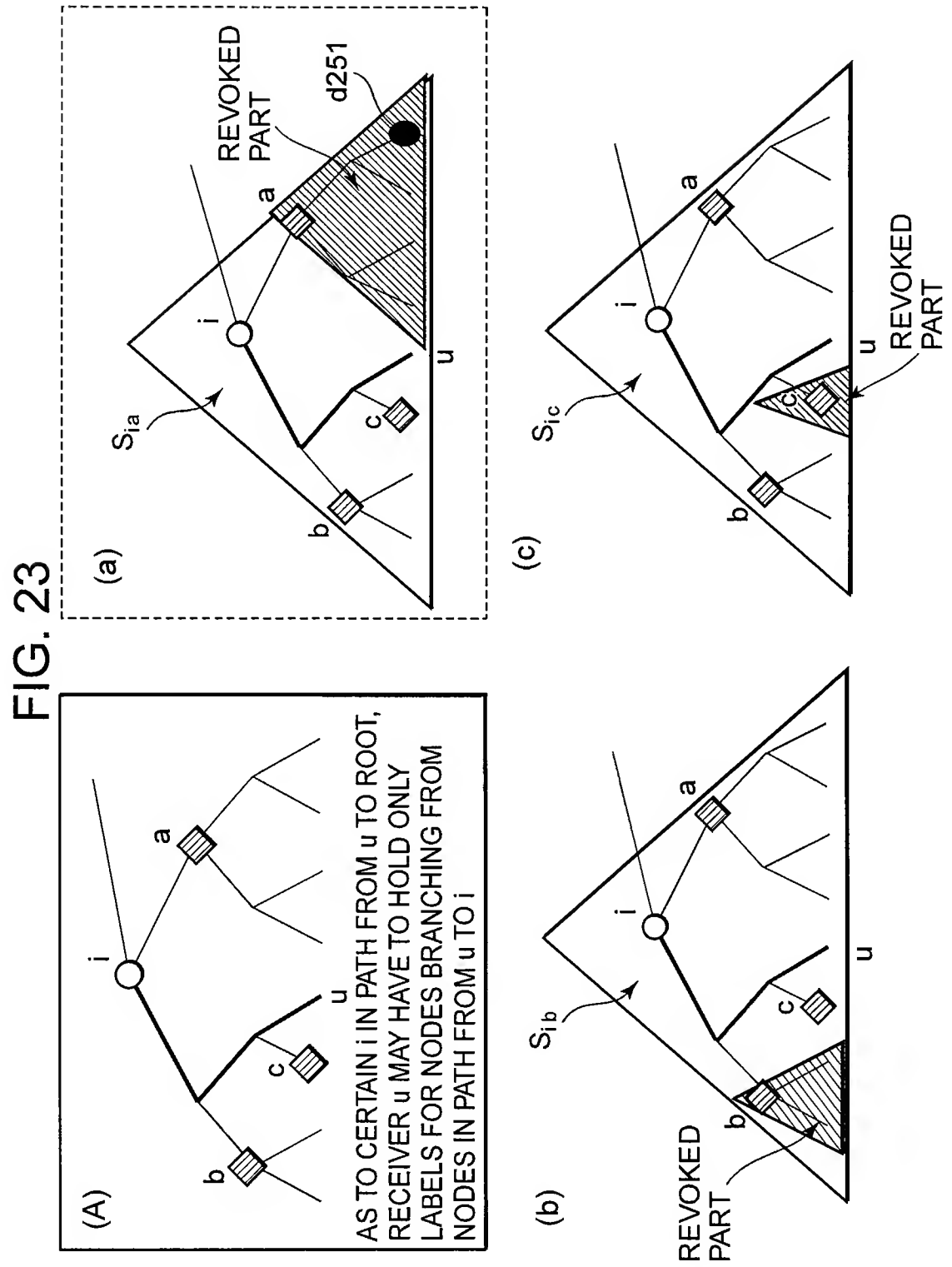
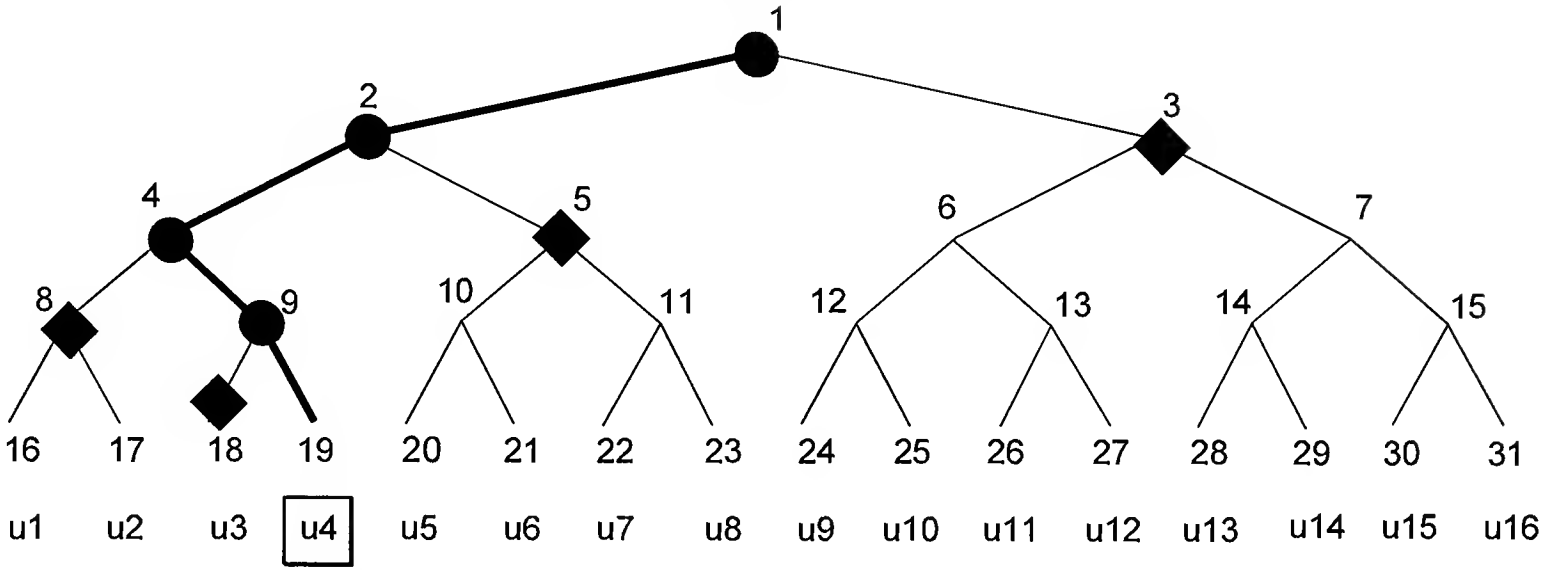

(A)
i
a
b
c
u
AS TO CERTAIN i IN PATH FROM u TO ROOT, RECEIVER u MAY HAVE TO HOLD ONLY LABELS FOR NODES BRANCHING FROM NODES IN PATH FROM u TO i
(a)
S_{ia}
REVOKED PART
d251
(b)
S_{ib}
REVOKED PART
(c)
S_{ic}
REVOKED PART

FIG. 24



LABEL OWNED BY u4

- j = 3, 5, 8, 18 FOR i = 1
- j = 5, 8, 18 FOR i = 2
- j = 8, 18 FOR i = 4
- j = 18 FOR i = 9
- ONE LABEL IN CASE OF NO REVOCATION

NUMBER OF LABELS HELD BY RECEIVER (INCLUDING ONE USED IF NONE ARE REVOKED)

$$1+\sum_{k=1}^{\log N} k = \frac{1}{2}\log^2 N + \frac{1}{2}\log N + 1$$

FIG. 25

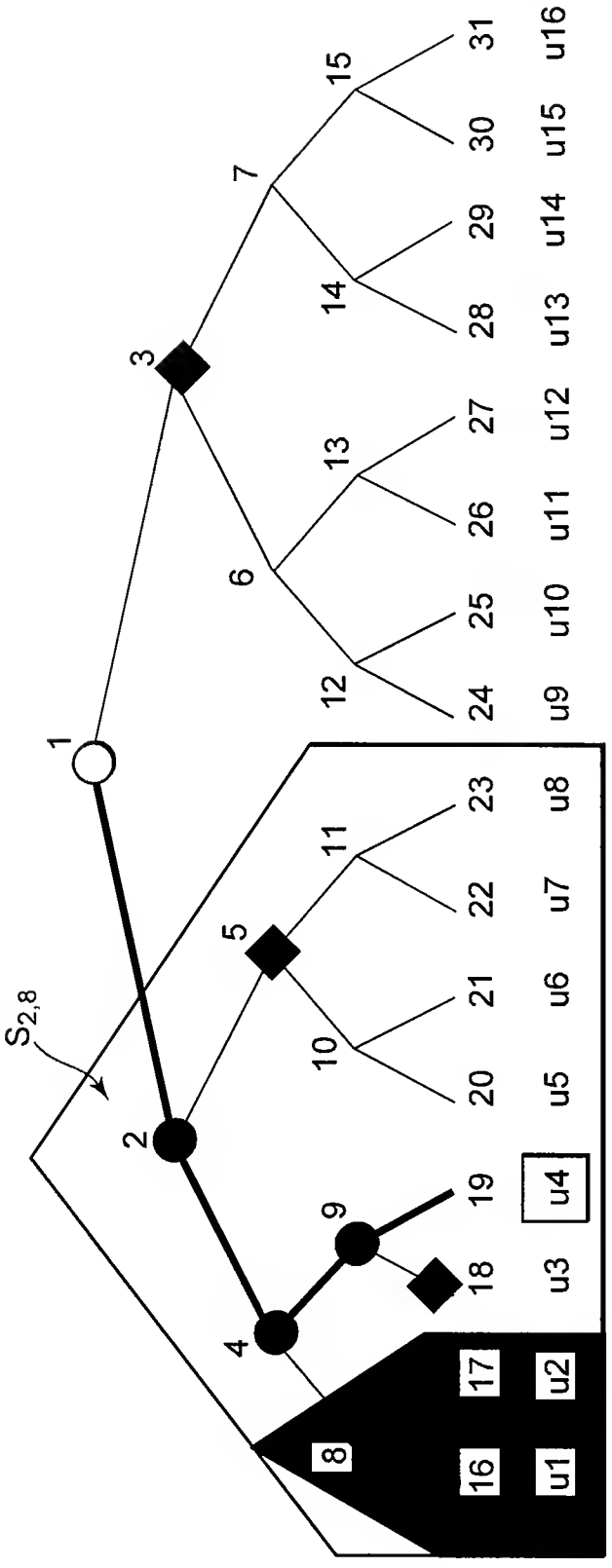

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
u1
u2
u3
u4
u5
u6
u7
u8
u9
u10
u11
u12
u13
u14
u15
u16
$S_{2,8}$

FIG. 26

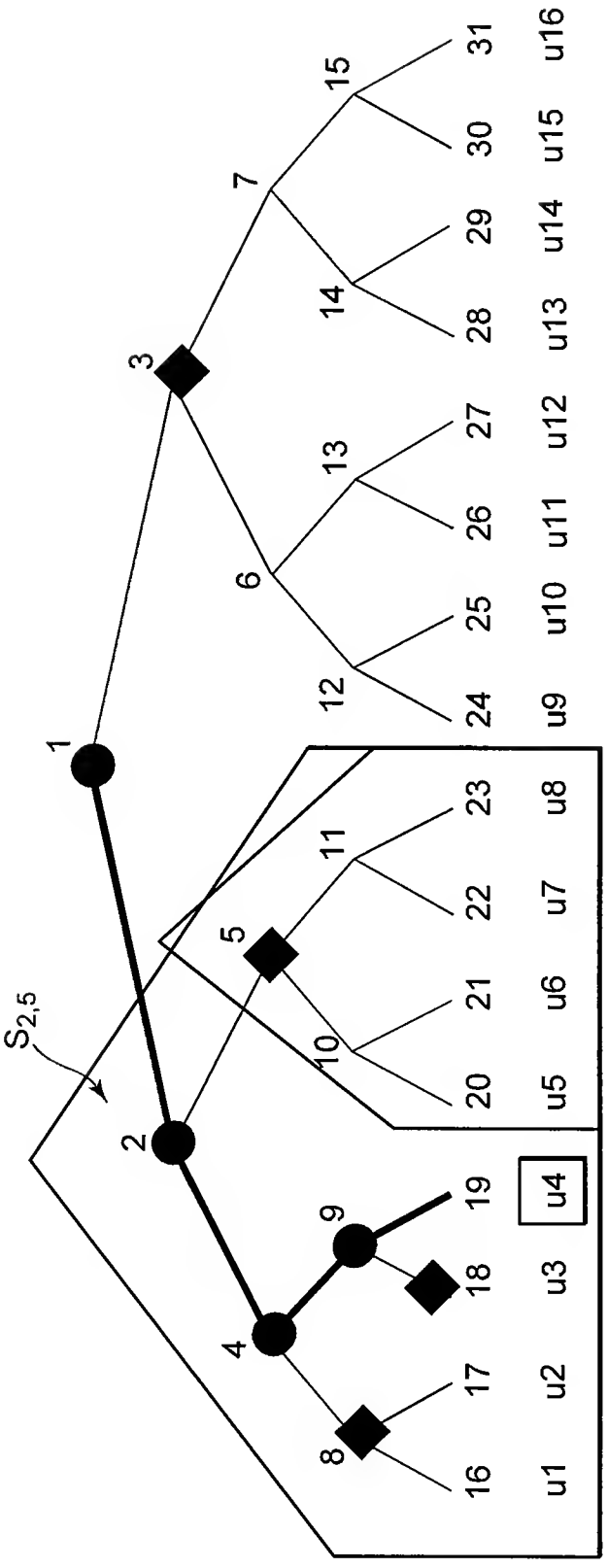

$S_{2,5}$
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
u1
u2
u3
u4
u5
u6
u7
u8
u9
u10
u11
u12
u13
u14
u15
u16

FIG. 27

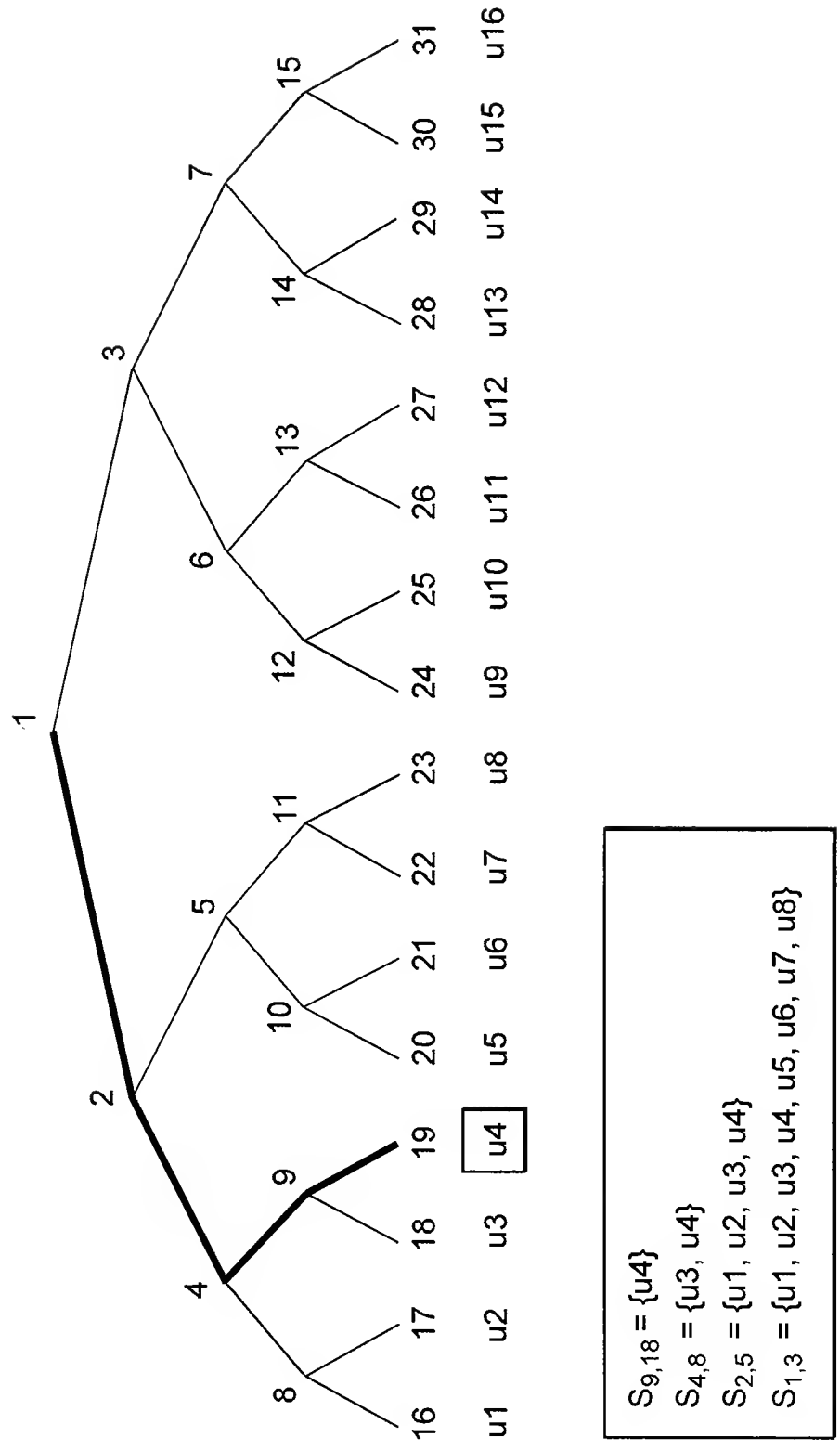
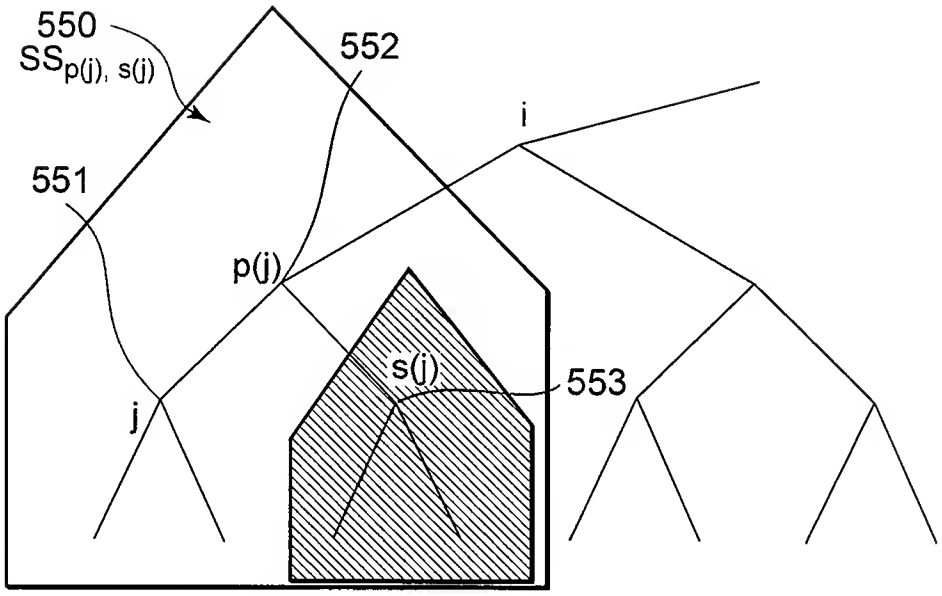

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
u1 u2 u3 u4 u5 u6 u7 u8 u9 u10 u11 u12 u13 u14 u15 u16
$S_{9,18}$ = {u4}
$S_{4,8}$ = {u3, u4}
$S_{2,5}$ = {u1, u2, u3, u4}
$S_{1,3}$ = {u1, u2, u3, u4, u5, u6, u7, u8}

FIG. 28



$NV_j = IL_{P(j),\ S(j)}$

$LABEL_{P(j),\ S(j)} = Hc\ (IL_{P(j),S(j)})$

$LABEL_{ij} = Hc\ (IL_{ij})$

FIG. 29

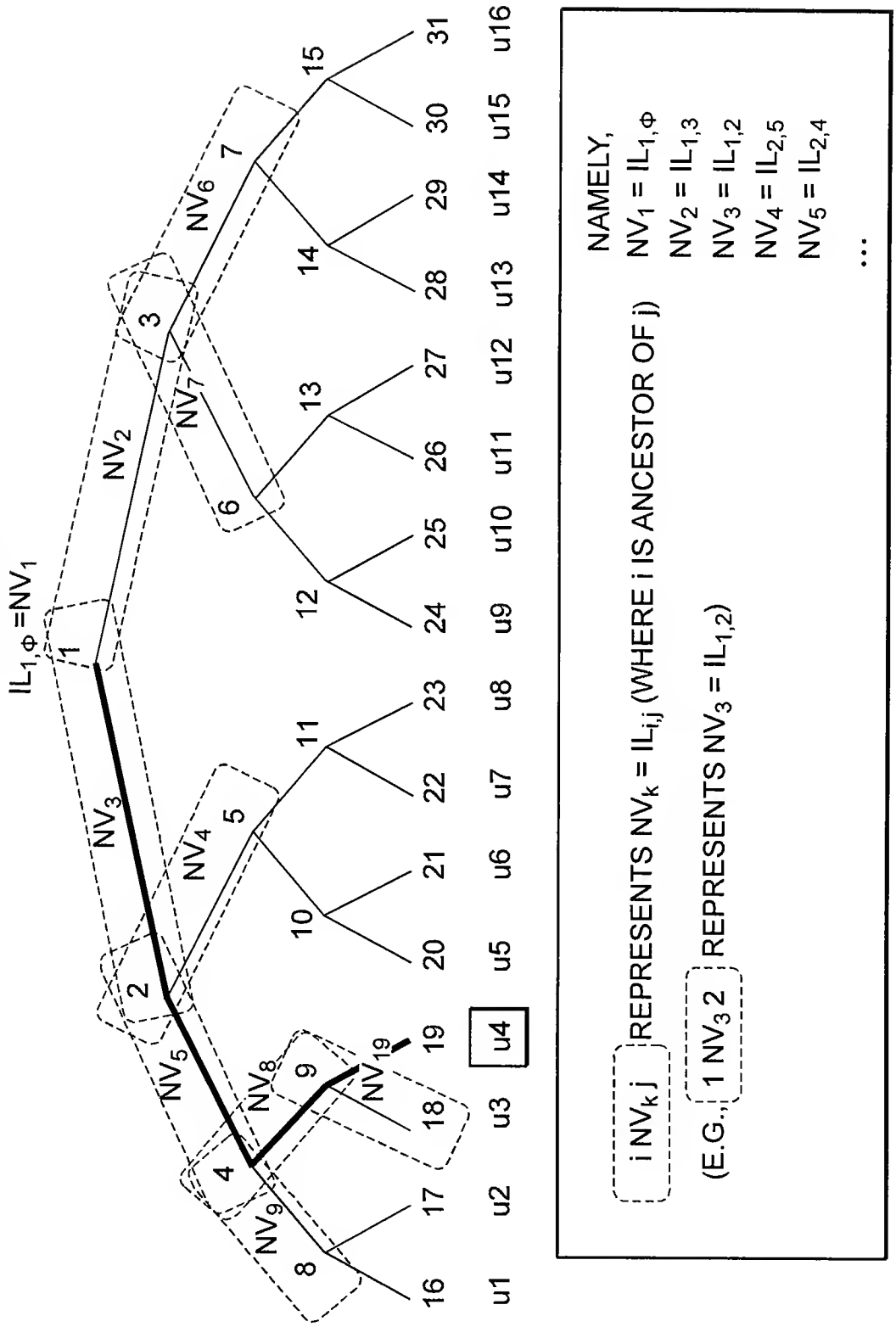

$IL_{1,\Phi}=NV_1$
NV_2
NV_3
NV_4
NV_5
NV_6
NV_7
NV_8
NV_9
NV_{19}
u1 u2 u3 u4 u5 u6 u7 u8 u9 u10 u11 u12 u13 u14 u15 u16
i NV_k j
REPRESENTS $NV_k = IL_{i,j}$ (WHERE i IS ANCESTOR OF j)
(E.G., 1 NV_3 2 REPRESENTS $NV_3 = IL_{1,2}$)
NAMELY,
$NV_1 = IL_{1,\Phi}$
$NV_2 = IL_{1,3}$
$NV_3 = IL_{1,2}$
$NV_4 = IL_{2,5}$
$NV_5 = IL_{2,4}$
...

FIG. 30

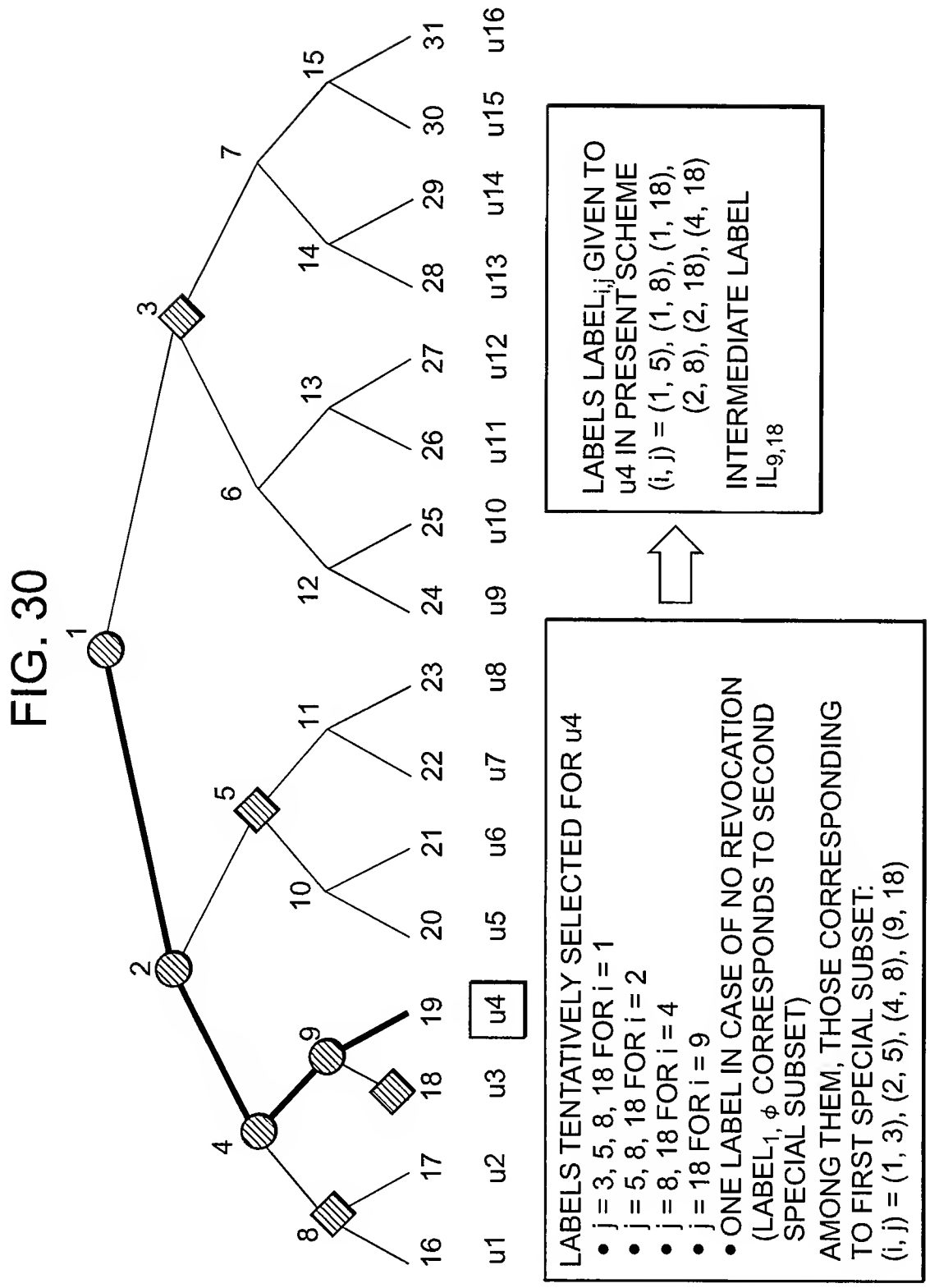

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
u1
u2
u3
u4
u5
u6
u7
u8
u9
u10
u11
u12
u13
u14
u15
u16
LABELS TENTATIVELY SELECTED FOR u4
• j = 3, 5, 8, 18 FOR i = 1
• j = 5, 8, 18 FOR i = 2
• j = 8, 18 FOR i = 4
• j = 18 FOR i = 9
• ONE LABEL IN CASE OF NO REVOCATION (LABEL1, ϕ CORRESPONDS TO SECOND SPECIAL SUBSET)
AMONG THEM, THOSE CORRESPONDING TO FIRST SPECIAL SUBSET:
(i, j) = (1, 3), (2, 5), (4, 8), (9, 18)
LABELS LABELi,j GIVEN TO u4 IN PRESENT SCHEME
(i, j) = (1, 5), (1, 8), (1, 18), (2, 8), (2, 18), (4, 18)
INTERMEDIATE LABEL
IL9,18

FIG. 31

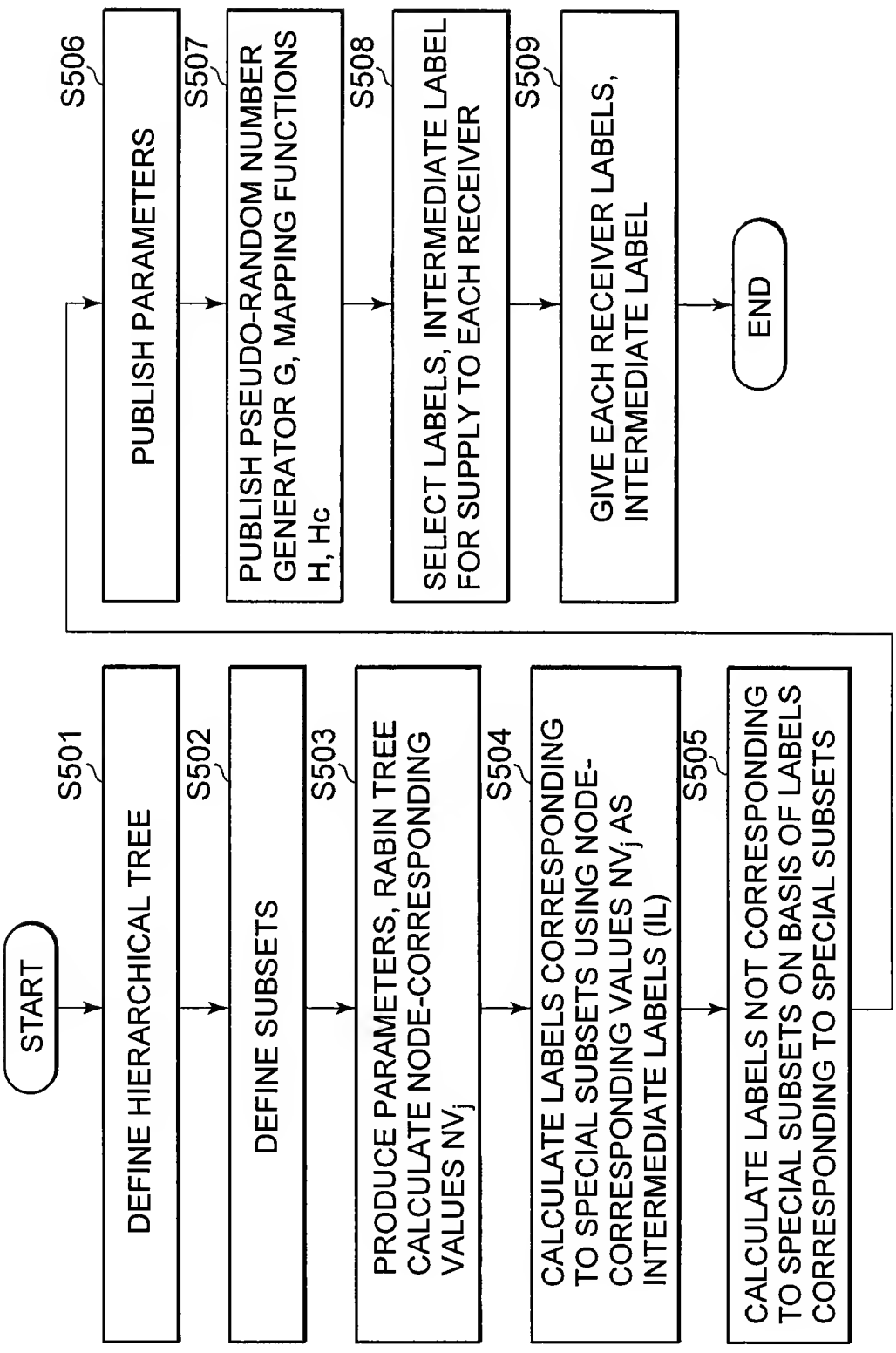

START
S501
DEFINE HIERARCHICAL TREE
S502
DEFINE SUBSETS
S503
PRODUCE PARAMETERS, RABIN TREE
CALCULATE NODE-CORRESPONDING VALUES NVj
S504
CALCULATE LABELS CORRESPONDING TO SPECIAL SUBSETS USING NODE-CORRESPONDING VALUES NVj AS INTERMEDIATE LABELS (IL)
S505
CALCULATE LABELS NOT CORRESPONDING TO SPECIAL SUBSETS ON BASIS OF LABELS CORRESPONDING TO SPECIAL SUBSETS
S506
PUBLISH PARAMETERS
S507
PUBLISH PSEUDO-RANDOM NUMBER GENERATOR G, MAPPING FUNCTIONS H, Hc
S508
SELECT LABELS, INTERMEDIATE LABEL FOR SUPPLY TO EACH RECEIVER
S509
GIVE EACH RECEIVER LABELS, INTERMEDIATE LABEL
END

FIG. 32

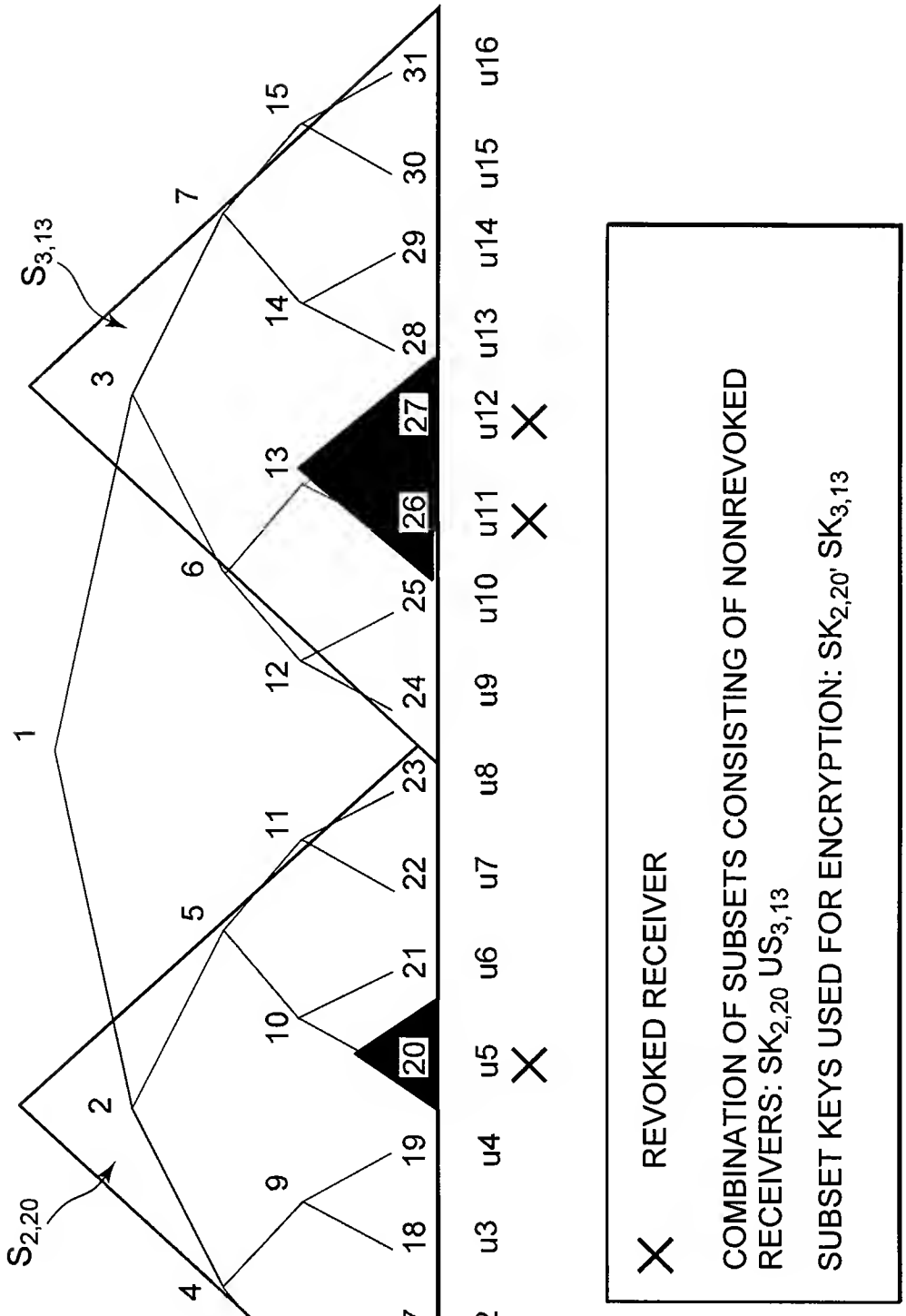

$S_{2,20}$
$S_{3,13}$
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
u1
u2
u3
u4
u5
u6
u7
u8
u9
u10
u11
u12
u13
u14
u15
u16
X
REVOKED RECEIVER
COMBINATION OF SUBSETS CONSISTING OF NONREVOKED RECEIVERS: $SK_{2,20}$ $US_{3,13}$
SUBSET KEYS USED FOR ENCRYPTION: $SK_{2,20'}$ $SK_{3,13}$

FIG. 33

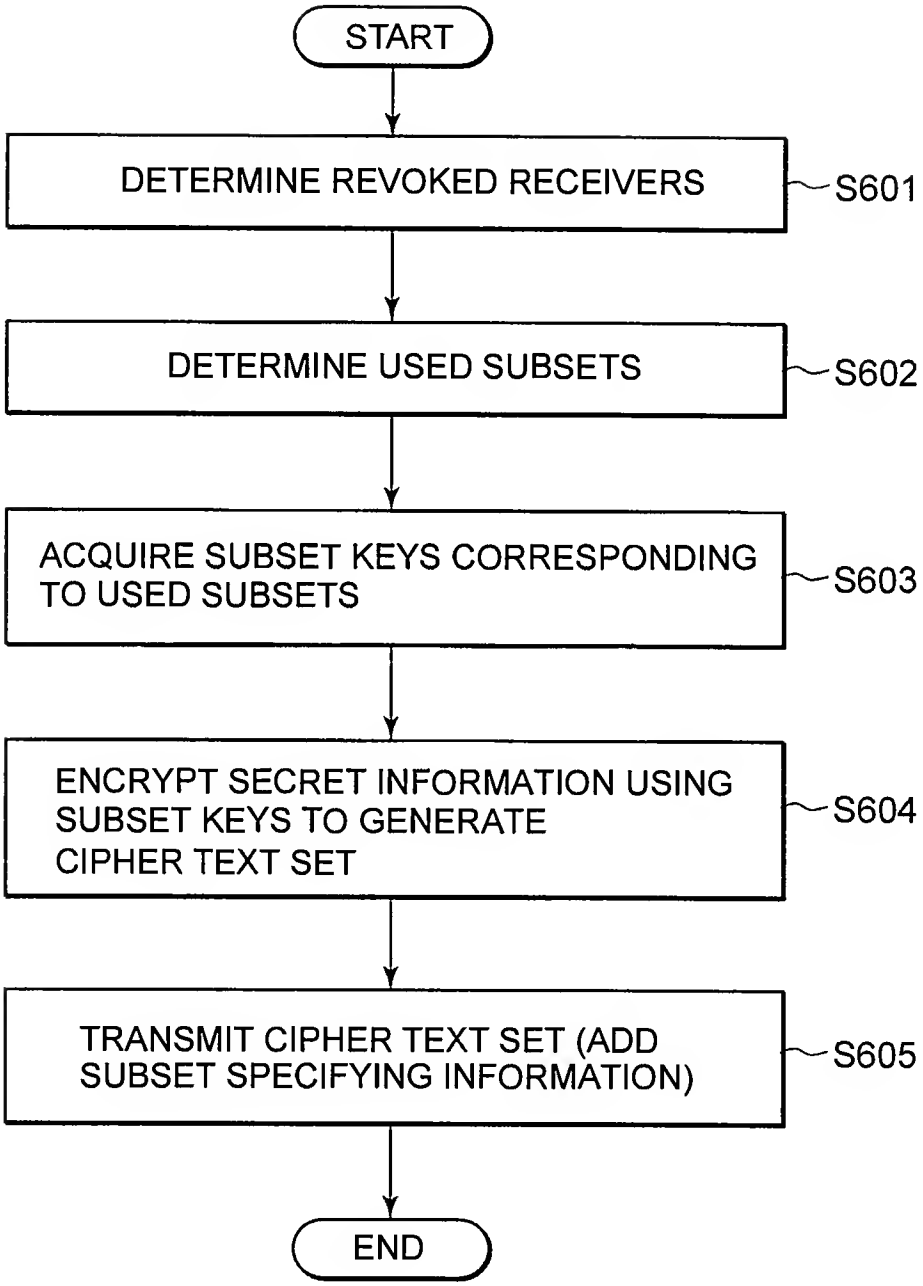

START
DETERMINE REVOKED RECEIVERS
S601
DETERMINE USED SUBSETS
S602
ACQUIRE SUBSET KEYS CORRESPONDING TO USED SUBSETS
S603
ENCRYPT SECRET INFORMATION USING SUBSET KEYS TO GENERATE CIPHER TEXT SET
S604
TRANSMIT CIPHER TEXT SET (ADD SUBSET SPECIFYING INFORMATION)
S605
END

FIG. 34

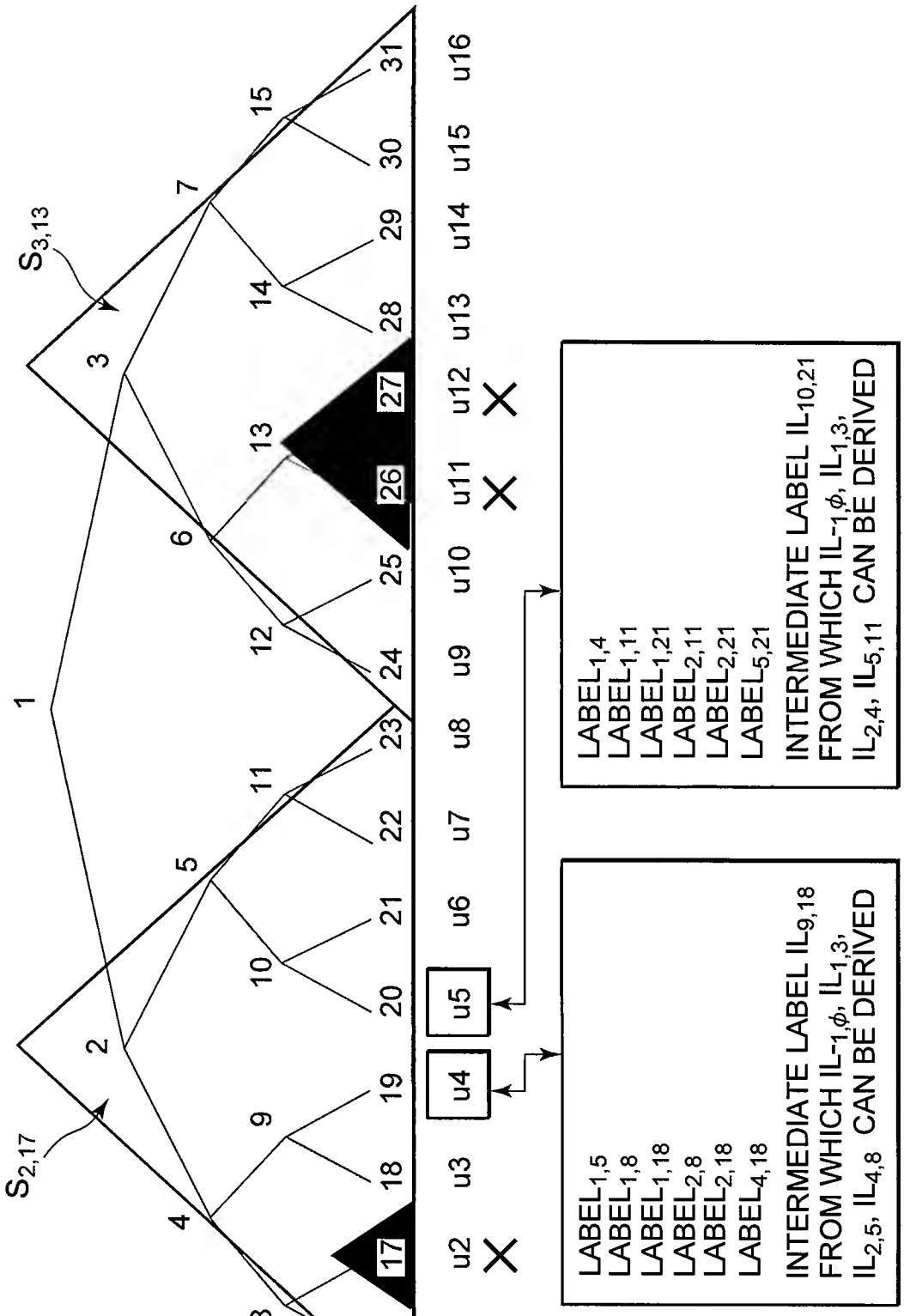

$S_{2,17}$
$S_{3,13}$
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
u1
u2
u3
u4
u5
u6
u7
u8
u9
u10
u11
u12
u13
u14
u15
u16
$LABEL_{1,5}$
$LABEL_{1,8}$
$LABEL_{1,18}$
$LABEL_{2,8}$
$LABEL_{2,18}$
$LABEL_{4,18}$
INTERMEDIATE LABEL $IL_{9,18}$
FROM WHICH $IL_{-1,\phi}$, $IL_{1,3}$,
$IL_{2,5}$, $IL_{4,8}$ CAN BE DERIVED
$LABEL_{1,4}$
$LABEL_{1,11}$
$LABEL_{1,21}$
$LABEL_{2,11}$
$LABEL_{2,21}$
$LABEL_{5,21}$
INTERMEDIATE LABEL $IL_{10,21}$
FROM WHICH $IL_{-1,\phi}$, $IL_{1,3}$,
$IL_{2,4}$, $IL_{5,11}$ CAN BE DERIVED

FIG. 35

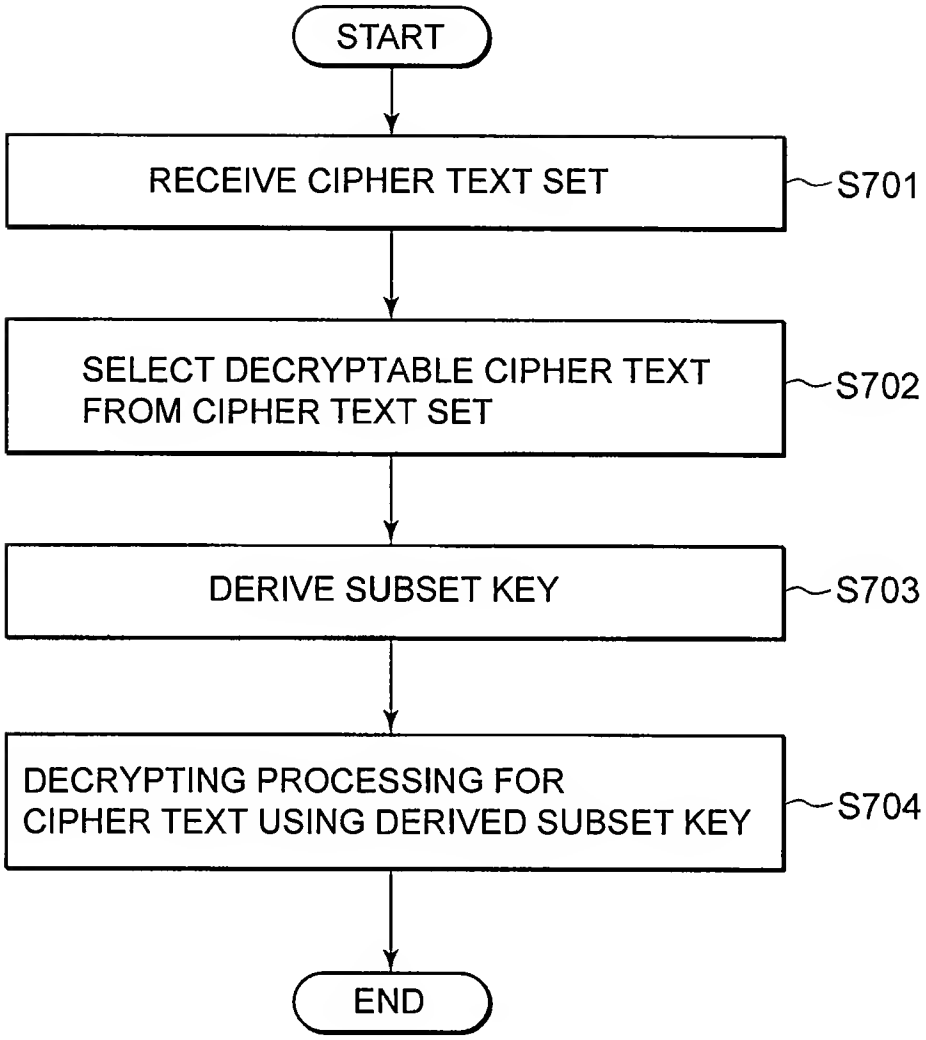

START
RECEIVE CIPHER TEXT SET
S701
SELECT DECRYPTABLE CIPHER TEXT FROM CIPHER TEXT SET
S702
DERIVE SUBSET KEY
S703
DECRYPTING PROCESSING FOR CIPHER TEXT USING DERIVED SUBSET KEY
S704
END

FIG. 36

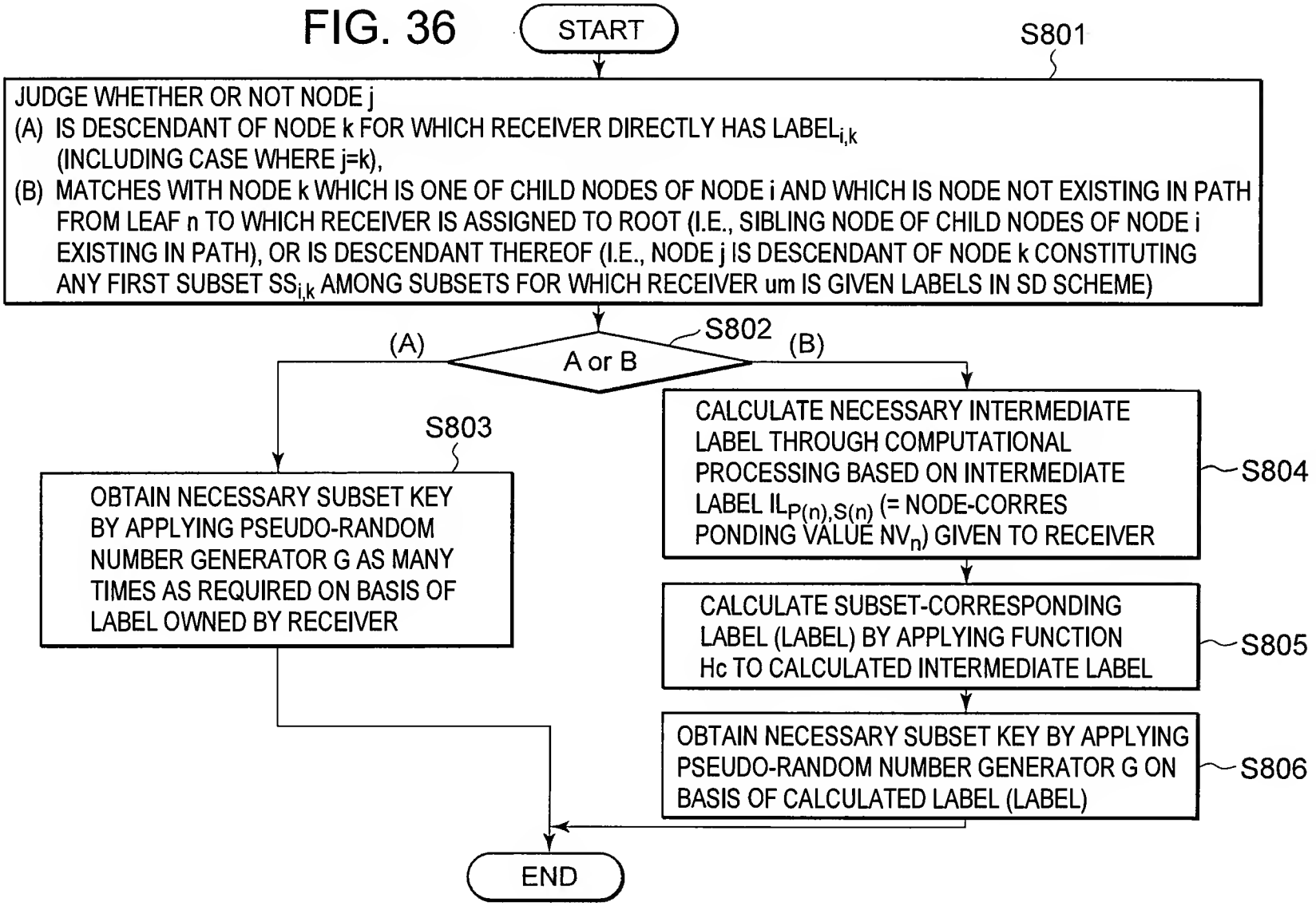

START
S801
JUDGE WHETHER OR NOT NODE j
(A) IS DESCENDANT OF NODE k FOR WHICH RECEIVER DIRECTLY HAS LABEL$_{i,k}$ (INCLUDING CASE WHERE j=k),
(B) MATCHES WITH NODE k WHICH IS ONE OF CHILD NODES OF NODE i AND WHICH IS NODE NOT EXISTING IN PATH FROM LEAF n TO WHICH RECEIVER IS ASSIGNED TO ROOT (I.E., SIBLING NODE OF CHILD NODES OF NODE i EXISTING IN PATH), OR IS DESCENDANT THEREOF (I.E., NODE j IS DESCENDANT OF NODE k CONSTITUTING ANY FIRST SUBSET SS$_{i,k}$ AMONG SUBSETS FOR WHICH RECEIVER um IS GIVEN LABELS IN SD SCHEME)
S802
A or B
(A)
(B)
S803
OBTAIN NECESSARY SUBSET KEY BY APPLYING PSEUDO-RANDOM NUMBER GENERATOR G AS MANY TIMES AS REQUIRED ON BASIS OF LABEL OWNED BY RECEIVER
CALCULATE NECESSARY INTERMEDIATE LABEL THROUGH COMPUTATIONAL PROCESSING BASED ON INTERMEDIATE LABEL IL$_{P(n),S(n)}$ (= NODE-CORRESPONDING VALUE NV$_n$) GIVEN TO RECEIVER
S804
CALCULATE SUBSET-CORRESPONDING LABEL (LABEL) BY APPLYING FUNCTION Hc TO CALCULATED INTERMEDIATE LABEL
S805
OBTAIN NECESSARY SUBSET KEY BY APPLYING PSEUDO-RANDOM NUMBER GENERATOR G ON BASIS OF CALCULATED LABEL (LABEL)
S806
END

FIG. 37

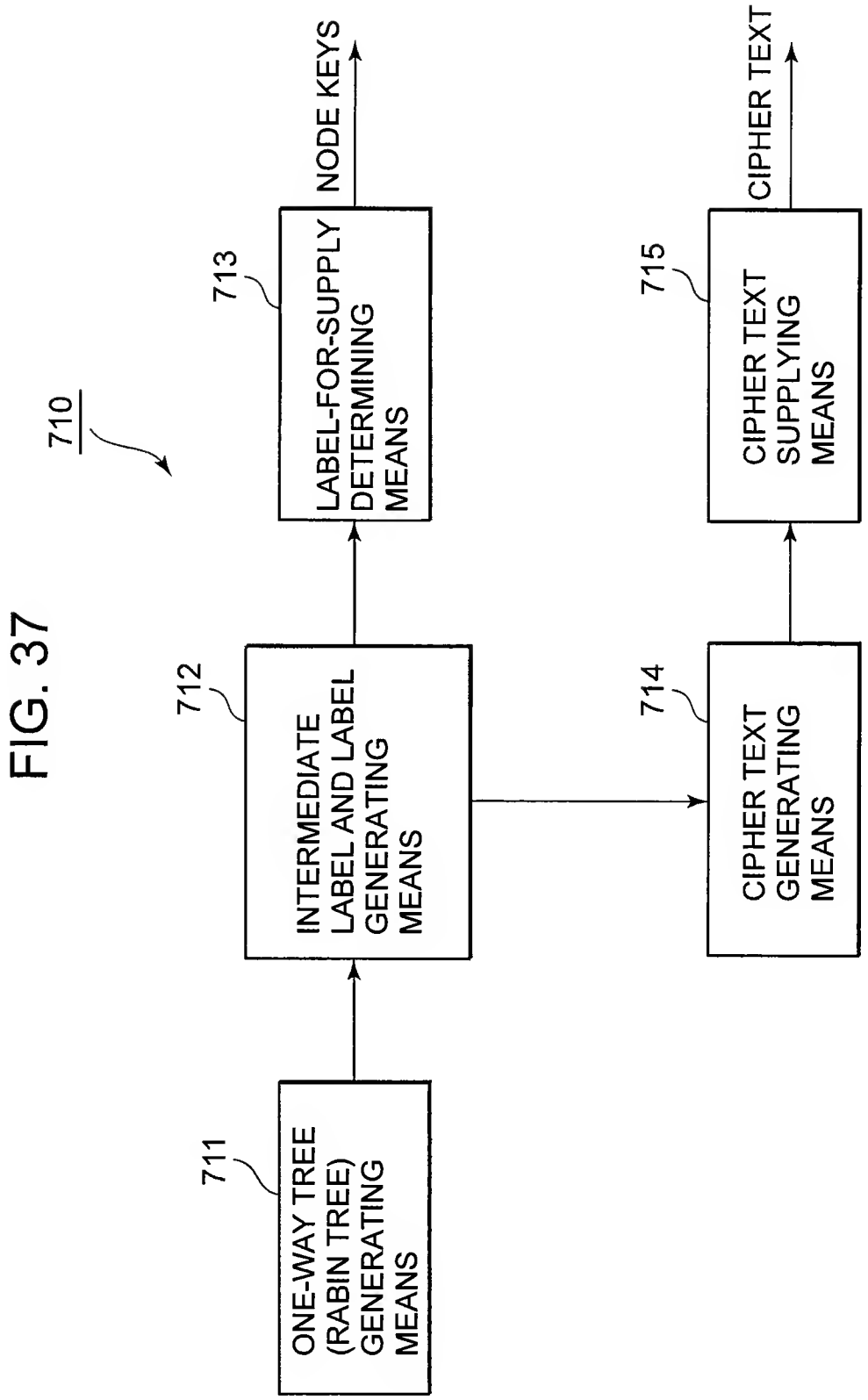

710
711
ONE-WAY TREE (RABIN TREE) GENERATING MEANS
712
INTERMEDIATE LABEL AND LABEL GENERATING MEANS
713
LABEL-FOR-SUPPLY DETERMINING MEANS
NODE KEYS
714
CIPHER TEXT GENERATING MEANS
715
CIPHER TEXT SUPPLYING MEANS
CIPHER TEXT

FIG. 38

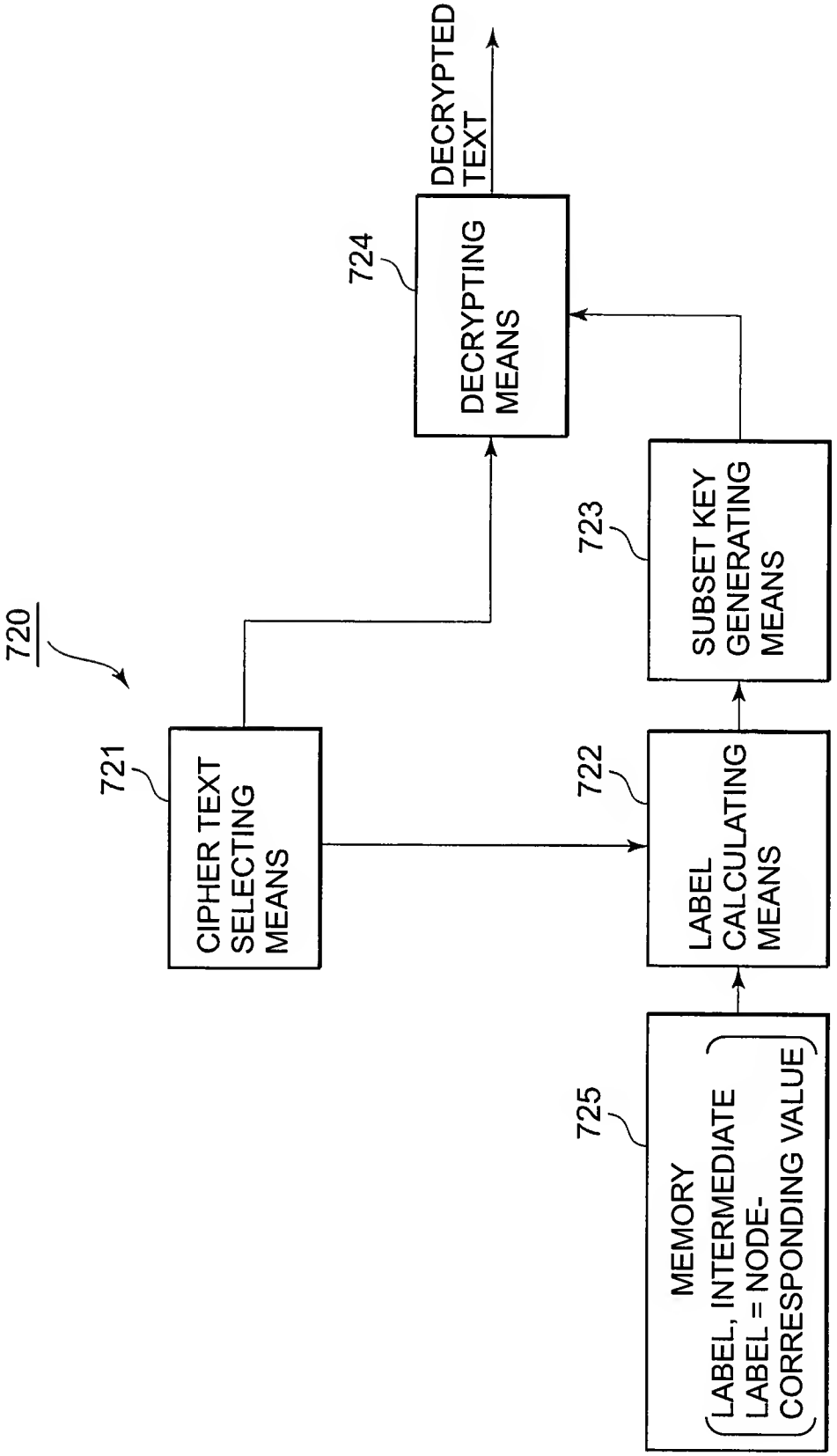

720
721
CIPHER TEXT SELECTING MEANS
722
LABEL CALCULATING MEANS
723
SUBSET KEY GENERATING MEANS
724
DECRYPTING MEANS
DECRYPTED TEXT
725
MEMORY
(LABEL, INTERMEDIATE LABEL = NODE-CORRESPONDING VALUE)

FIG. 39

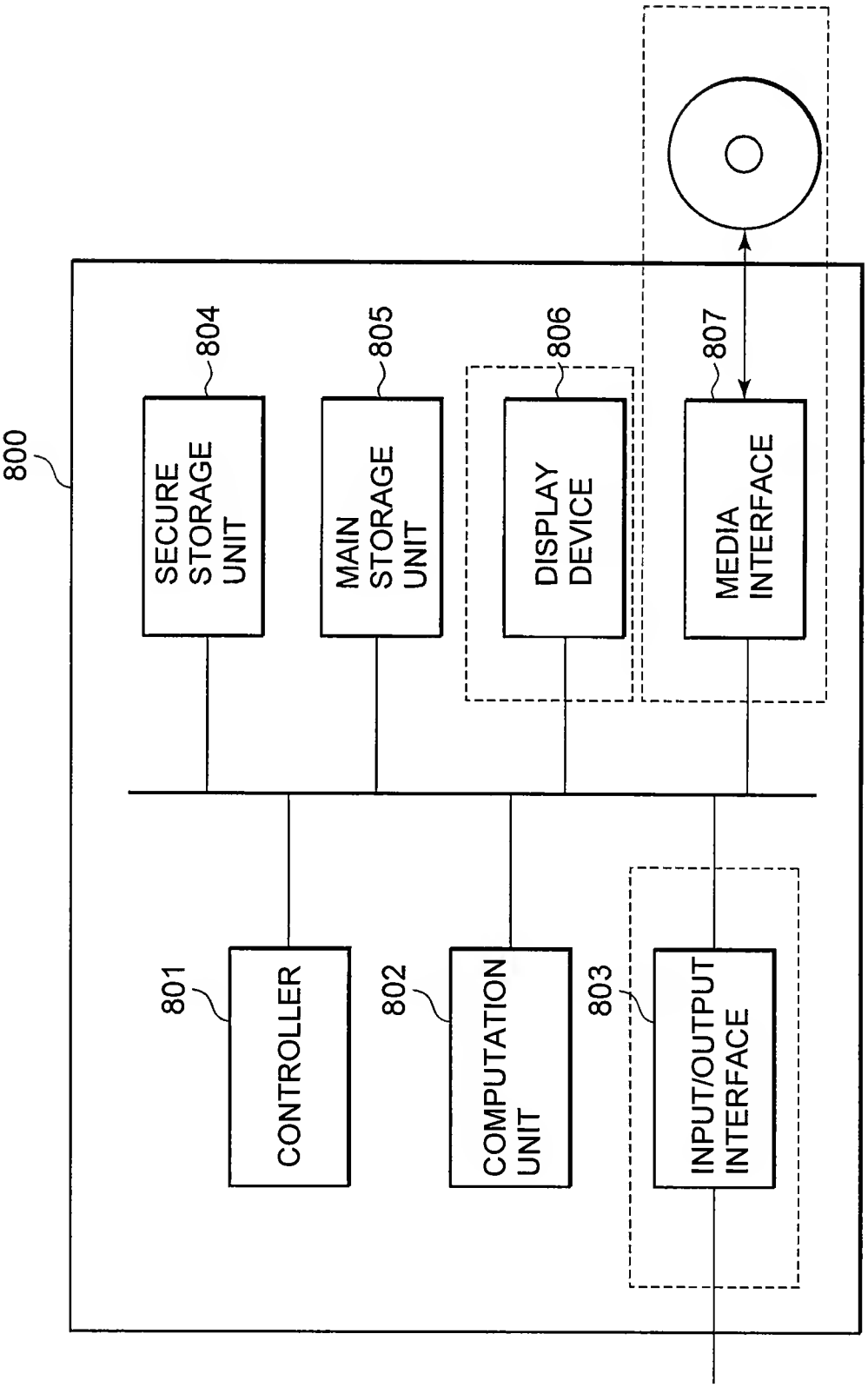
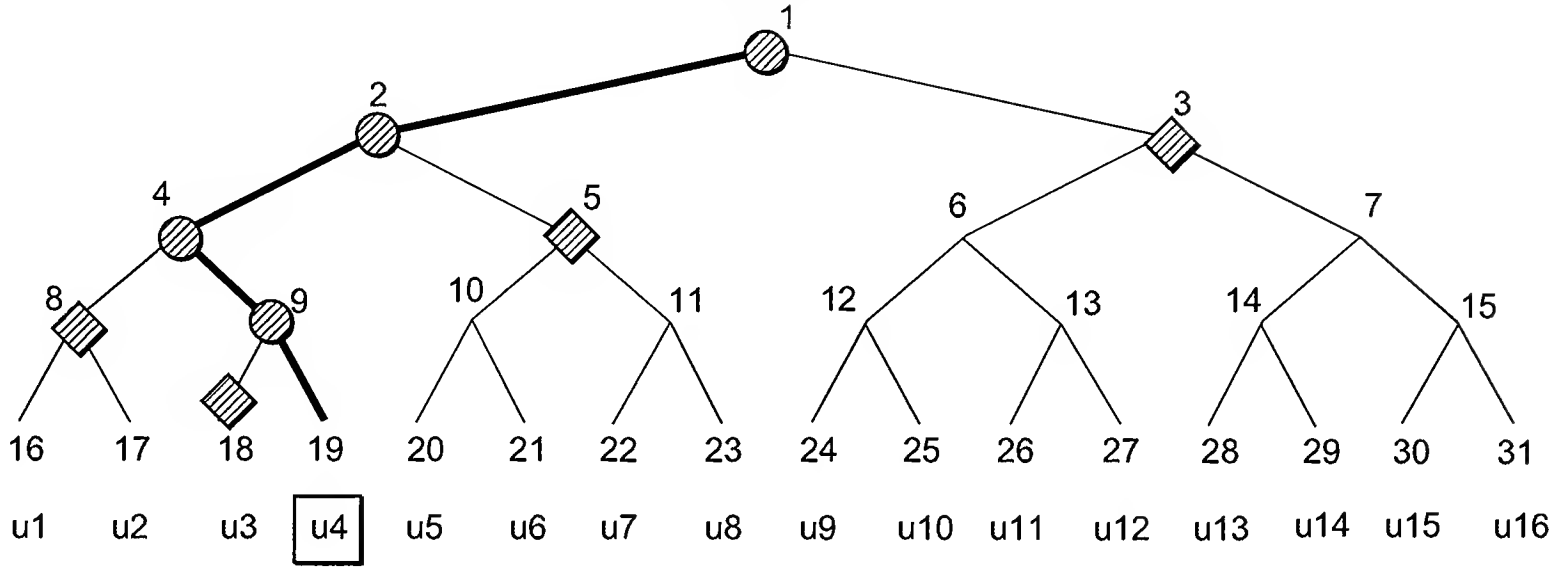

800
804
SECURE STORAGE UNIT
805
MAIN STORAGE UNIT
806
DISPLAY DEVICE
807
MEDIA INTERFACE
801
CONTROLLER
802
COMPUTATION UNIT
803
INPUT/OUTPUT INTERFACE

FIG. 40



LABELS TENTATIVELY SELECTED FOR u4

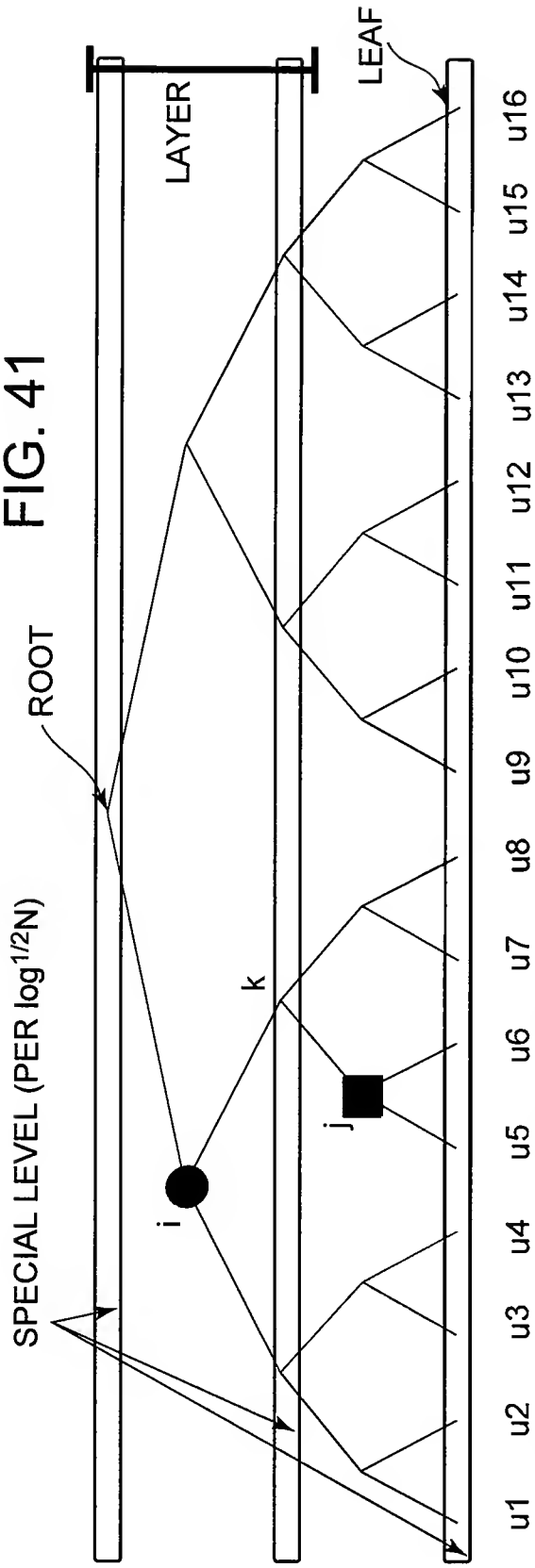
- j = 3, 5, 8, 18 FOR i = 1
- j = 5, 8, 18 FOR i = 2
- j = 8, 18 FOR i = 4
- j = 18 FOR i = 9
- ONE LABEL IN CASE OF NO REVOCATION ($LABEL_{1,\phi}$ CORRESPONDS TO SECOND SPECIAL SUBSET)

AMONG THEM, THOSE CORRESPONDING TO FIRST SPECIAL SUBSET:
(i, j) = (1, 3), (2, 5), (4, 8), (9, 18)

LABELS $LABEL_{i,j}$ GIVEN TO u4 IN PRESENT SCHEME
(i, j) = (1, 5), (1, 8), (1, 18), (2, 8), (2, 18), (4, 18)

INTERMEDIATE LABEL
$IL_{9,18}$

FIG. 41



AMONG ALL SUBSET DIFFERENCE SETS $S_{i,j}$,
ANY SET SATISFYING AT LEAST ONE OF CONDITIONS

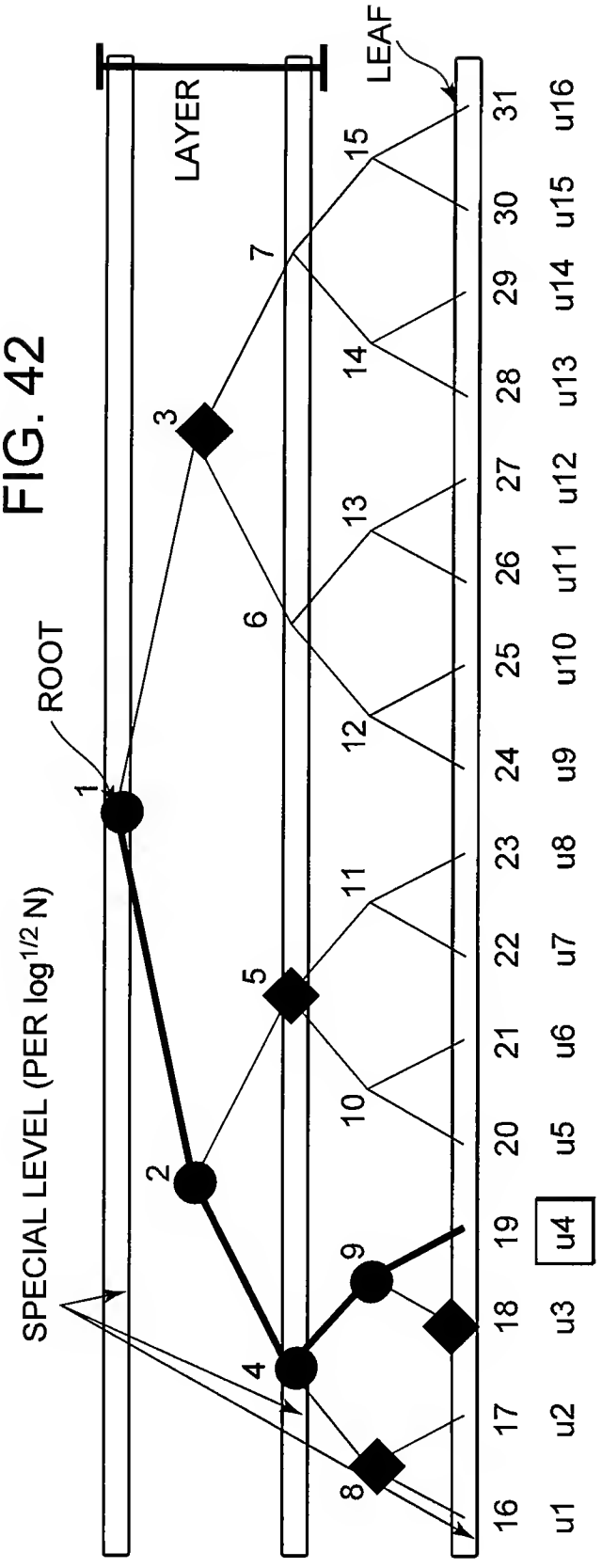
- BOTH i AND j BELONG TO SAME LAYER
- i IS AT SPECIAL LEVEL

IS DEFINED

IN ABOVE EXAMPLE, $S_{i,j}$ IS NOT DEFINED.
IT IS REPRESENTED AS UNION OF TWO SETS, SUCH AS
$S_{i,j} = S_{i,k} \cup S_{k,j}$
→ AMOUNTS OF COMMUNICATION DATA DOUBLES AT MAXIMUM COMPARED TO THAT IN SD

ONE KIND OF SPECIAL LEVEL IN Basic LSD
PLURAL KINDS OF SPECIAL LEVELS IN General LSD

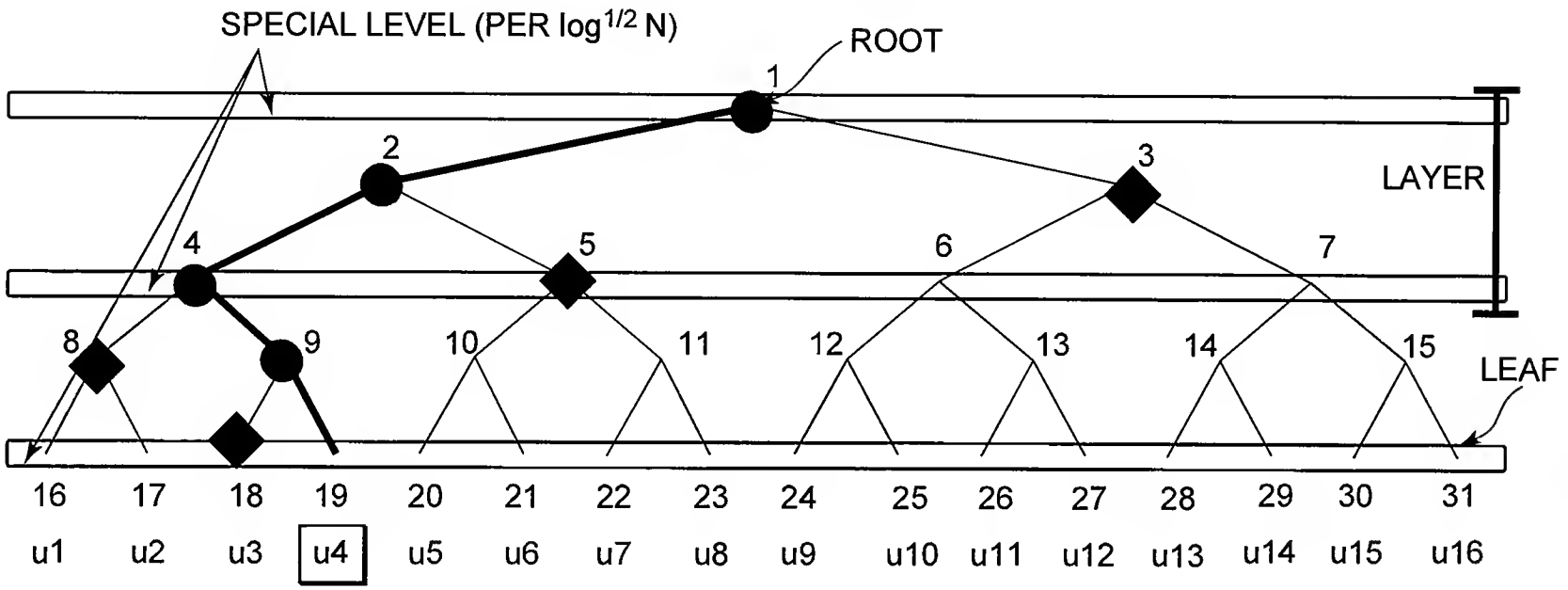
FIG. 42



LABELS OWNED BY u4
- j = 3, 5, 8, 18 FOR i = 1
- j = 5 FOR i = 2
- j = 8, 18 FOR i = 4
- j = 18 FOR i = 9
- ONE LABEL ($LABEL_{1,\phi}$) IN CASE OF NO REVOCATION

NUMBER OF LABELS HELD BY RECEIVER (INCLUDING ONE USED WHERE NONE ARE REVOKED) $\log^{3/2} N+1$

FIG. 43



LABELS OWNED BY u4

- j = 3, 5, 8, 18 FOR i = 1
- j = 5 FOR i = 2
- j = 8, 18 FOR i = 4
- j = 18 FOR i = 9
- ONE LABEL ($LABEL_{1,\phi}$) IN CASE OF NO REVOCATION

LABELS, INTERMEDIATE LABEL GIVEN TO u4 IN PRESENT SCHEME

(a) LABELS $LABEL_{i,j}$
(i, j) = (1, 5), (1, 8), (1, 18), (4, 18)

(b) INTERMEDIATE LABEL (NODE-CORRESPONDING VALUE)
$IL_{9,18} = (NV_{19})$